Software Reference for AT-x510 Series Switches

AlliedWare Plus™ Operating System Version 5.4.2A

AT-x510-28GTX
AT-x510-52GTX

C613-50023-01-REV A

Allied Telesis

Acknowledgments

This product includes software developed by the University of California, Berkeley and its contributors.
Copyright ©1982, 1986, 1990, 1991, 1993 The Regents of the University of California.
All rights reserved.

This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (http://www.openssl.org/).
Copyright ©1998-2008 The OpenSSL Project. All rights reserved.

This product includes software licensed under the GNU General Public License available from: http://www.gnu.org/licenses/gpl2.html

Source code for all GPL licensed software in this product can be obtained from the Allied Telesis GPL Code Download Center at:
http://www.alliedtelesis.com/support/default.aspx

Allied Telesis is committed to meeting the requirements of the open source licenses including the GNU General Public License (GPL) and will make all required source code available.

If you would like a copy of the GPL source code contained in Allied Telesis products, please send us a request by registered mail including a check for US$15 to cover production and shipping costs and a CD with the GPL code will be mailed to you.

GPL Code Request
Allied Telesis Labs (Ltd)
PO Box 8011
Christchurch.
New Zealand

Getting the most from this manual

Although you can view this document using Acrobat version 5, to get the best from this manual, we recommend using Adobe Acrobat Reader version 8. You can download Acrobat Reader 8 free from http://www.adobe.com/.

Table of Contents

Part 1 Setting up the Switch

Part 2 Layer Two Switching

Part 3 Layer Three, Switching and Routing

Part 4 Multicast Applications

Part 5 Access and Security

Part 6 Network Availability

Part 7 Network Management

Part 8 Virtual Chassis Stacking

Appendix A: Command List

Appendix B: Changes in Version 5.4.2A-0.1

Appendix C: GUI Reference

Appendix D: Glossary

Part 1: Setting up the Switch

Chapter 1: Getting Started

Introduction

This chapter introduces a number of commonly-used management features of the AlliedWare Plus^{TM} Operating System (OS).

How to Login

Step 1: Set the console baud rate

The default baud rate is 9600.

By default the AlliedWare Plus^{TM} OS supports VT100 compatible terminals on the console port. This means that the terminal size is 80 columns by 24 rows.

Step 2: Login with manager/friend

The defaults are:

```
username: manager
password: friend
```

The switch logs you into User Exec mode. From User Exec mode, you can perform high-level diagnostics (some **show** commands, ping, traceroute etc), start sessions (Telnet, SSH), and change mode.

How to get Command Help

The following kinds of command help are available:

- lists of valid parameters with brief descriptions (the ? key)
- completion of keywords (the Tab key)
- error messages for incomplete or incorrect syntax

Command Abbreviations

The AlliedWare Plus^{TM} CLI contains a number of abbreviations for its commands. For example, the **show interface** command can be entered in the abbreviated form shown below:

`awplus#`
`sh in vlan100`
	sh in vlan100

`awplus#`
`configure terminal24`
	Enter the Global Configuration mode.

Viewing a List of Valid Parameters

To get syntax help, type ? (i.e. "space question mark") after:

- the prompt. This will list all commands available in the mode you are in.
- one or more parameters. This will list parameters that can come next in the partial command.
- one or more letters of a parameter. This will list matching parameters.

Note The AlliedWare Plus^{TM} OS only displays one screenful of text at a time, with the prompt "--More--" at the end of each screenful. Press the space bar to display the next screenful or the Q key to return to the command prompt.

Example

To see which commands are available in Privileged Exec mode, enter "?" at the Privileged Exec mode command prompt:

```
awplus# ?
```

This results in the following output:

Figure 1-1: Example output from the ? command

```
Exec commands:
activate        Activate a script
cd              Change the current working directory
clear           Reset functions
clock           Manage clock
configure       Enter configuration mode
copy            Copy from one file to another
debug           Debugging functions (see also 'undebug')
delete          Delete a file
dir             List the files on a filesystem
disable         Turn off privileged mode command
dot1x           IEEE 802.1X Port-Based Access Control
echo            Echo a string
edit            Text Editor
enable          Turn on privileged mode command
erase           Erase the system startup configuration
exit            End current mode and down to previous mode
help            Description of the interactive help system
license         Activate software feature license
logout          Exit from the EXEC
mail            Send an email
mkdir           Make a new directory
move            Rename or move a file
mstat           Show statistics after multiple multicast
                traceroutes
mtrace           Trace multicast path from source to destination
no              Negate a command or set its defaults
ping            Send echo messages
platform        Execute built-in self-tests
pwd             Print the current working directory
quit            Exit current mode and down to previous mode
reboot          Halt and perform a cold restart
reload          Halt and perform a cold restart
remote-command  Remote stack member command execution
  rmdir           Remove a directory
rmon            Debugging functions (see also 'undebug')
show            Show running system information
ssh             Open an SSH connection
tcpdump         Execute tcpdump
telnet          Open a telnet connection
terminal        Set terminal line parameters
test            Test device functionality
traceroute      Trace route to destination
trigger         Automatic scripted responses to device events
undebug         Disable debugging functions (see also 'debug')
wait            Wait for a specified number of seconds
write           Write running configuration to memory, file or
                terminal
```

Example To see which commands are available in Configuration mode, enter “?” at the Config mode command prompt:.

```
awplus# configure terminal
```

```
awplus(config)# ?
```

This results in the following output:

Figure 1-2: Example output from the ? command

```
Configure commands:
  aaa                  Authentication,Authorization and Accounting
  access-list          Add an access list entry
  arp                  Address Resolution Protocol (ARP)
  auth-web-server    Web authentication server configuration
                       commands
  banner               Define a login banner
    boot                 Boot configuration
  class-map            Class map command
  clock                Manage clock
  crypto               Security Specific Commands
  cvlan                Configure C-VLAN parameters
  debug                Debugging functions (see also 'undebug')
  default              Restore default settings
  do                   To run exec commands in config mode
  dot1x                IEEE 802.1X Port-Based Access Control
  enable               Modify enable password parameters
  epsr                 Ethernet Protection Switching Ring (EPSR)
  exception            Configure exception settings
  exit                 End current mode and down to previous mode
  fib                  FIB information
  help                 Description of the interactive help system
  hostname             Set system's network name
  interface            Select an interface to configure
  ip                   Internet Protocol (IP)
  ipv6                 Internet Protocol version 6 (IPv6)
  key                  Authentication key management
  lacp                 LACP commands
  line                 Configure a terminal line
  log                  Logging control
  loop-protection      Loop Protection
  mac                  mac address
  mail                 Send an email
  max-fib-routes       Set maximum fib routes number
  max-static-routes    Set maximum static routes number
  maximum-access-list  Maximum access-list entries
  maximum-paths        Set multipath numbers installed to FIB
  mls                  Multi-Layer Switch(L2/L3)
  no                   Negate a command or set its defaults
  ntp                  Configure NTP
    ping-poll            Ping Polling
  platform           Configure global settings for the switch
                       asic
  policy-map           Policy map command
  radius-server        Radius server
  remote-command       Remote stack member command execution
    rmon                 Remote Monitoring Protocol (RMON)
      service              Modify use of network based services
  show                 Show running system information
  snmp-server          Enable the snmp agent
  spanning-tree        Spanning tree commands
  ssh                  Secure Shell
  stack                Manage VCS feature
  system               System properties
  telnet               Configure telnet
  trigger            Automatic scripted responses to device
                       events
  undebug            Disable debugging functions (see also
                       'debug')
  username             Establish User Name Authentication
  virtual-server       Virtual-server configuration
  vlan                 Configure VLAN parameters
  vrrp                 VRRP configuration
```

Example

To see which **show** commands that start with "i" are available in Privileged Exec mode, enter "?" after **show i**:

```
awplus# show i?
```

This results in the following output:

Figure 1-3: Example output from the **show i?** command

```
interface          Select an interface to configure
  ip                 Internet Protoc6ol (IP)
  ipv6               Internet Protocol version 6 (IPv6)
```

Examples

To use the ? help to work out the syntax for the **clock timezone** command, enter the following sequence of commands:

```
awplus(config)# clock ?
```

```
summer-time  Manage summer-time
timezone     Set clock timezone
```

```
awplus(config)# cloc timezone ?
```

```
TIMEZONE  Timezone name, up to 5 characters
```

```
awplus(config)# clock timezone NZST ?
```

```
minus  negative offset
plus   positive offset
```

```
awplus(config)# clock timezone NZST plus ?
```

```
<0-12>  Time zone offset to UTC
```

```
awplus(config)# clock timezone NZST plus 12
```

The above example demonstrates that the ? help only indicates what you can type **next**. For commands that have a series of parameters, like **clock timezone**, the ? help does not make the number of parameters obvious.

Completing Keywords

To complete keywords, type the Tab key after part of the command.

If only one keyword matches the partial command, the AlliedWare Plus^{TM} OS fills in that keyword. If multiple keywords match, it lists them.

Examples In this example we use Tab completion in successive steps to build the complete command **show ip dhcp server summary**. We have included "<Tab>" to show where to type the Tab key - this is not displayed on screen.

```
awplus# show ip <Tab>
```

Figure 1-4: Example output after entering the command, **show ip** <Tab>

```
dhcp                 dhcp-relay           domain-list
domain-name          extcommunity-list    filter
forwarding           igmp                 interface
irdp                 mroute               mvif
name-server          nat
pim                  protocols            rip
route                rpf
```

```
awplus# show ip d<Tab>
```

Figure 1-5: Example output after entering the command, **show ip d**<Tab>

```
dhcp         dhcp-relay       domain-list       domain-name
```

```
awplus# show ip dhcp <Tab>
```

Figure 1-6: Example output from the **show ip dhcp** <Tab> command

```
binding  pool        server
```

```
awplus# show ip dhcp server s<Tab>
```

Figure 1-7: Example output from the **show ip dhcp s**<Tab> command

```
statistics           summary
```

Viewing Command Error Messages

The switch displays the following generic error messages about command input:

% Incomplete command—this message indicates that the command requires more parameters. Use the ? help to find out what other parameters are available.

```
awplus# interface
```

```
% Incomplete command.
```

% Invalid input detected at '^' marker—this indicates that the switch could not process the command you entered. The switch also prints the command and marks the first invalid character by putting a '^' under it. Note that you may get this error if you enter a command in the wrong mode, as the following output shows.

```
awplus# interface port1.0.1
```

```
interface port1.0.1
 ^
% Invalid input detected at '^' marker.
```

% Unrecognized command—when you try to use ? help and get this message, it indicates that the switch can not provide help on the command because it does not recognize it. This means the command does not exist, or that you have entered it in the wrong mode, as the following output shows.

```
awplus# interface ?
```

```
% Unrecognized command
```

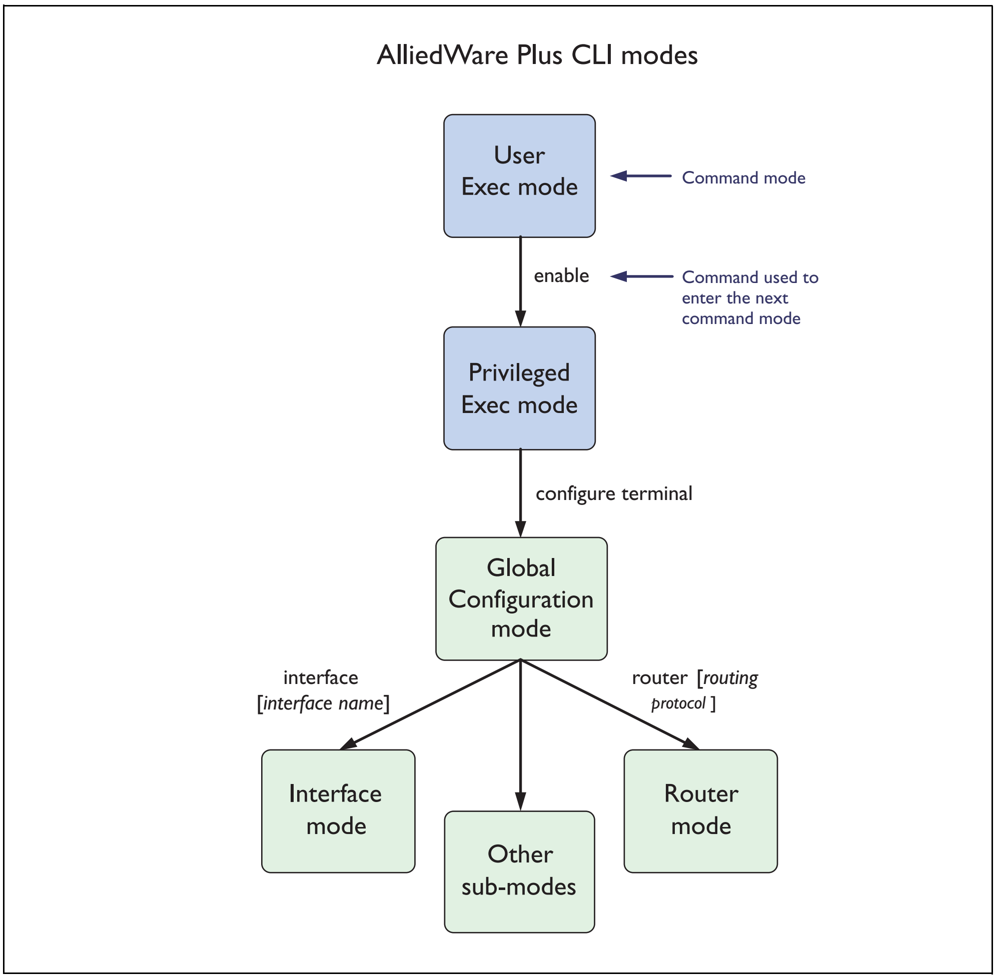
Note The AlliedWare Plus^{TM} OS does not tell you when commands are successful. If it does not display an error message, you can assume the command was successful.

How to Work with Command Modes

The following figure shows the command mode hierarchy and the commands you use to move to lower-level modes.

Multiple users can telnet and issue commands using the User Exec mode and the Privileged Exec mode. However, only one user is allowed to use the Configure mode at a time. This prevents multiple users from issuing configuration commands simultaneously.

Figure 1-8: AlliedWare Plus^{TM} CLI modes


AlliedWare Plus CLI modes
User Exec mode
Command mode
enable
Command used to enter the next command mode
Privileged Exec mode
configure terminal
Global Configuration mode
interface [interface name]
router [routing protocol]
Interface mode
Other sub-modes
Router mode

User Exec mode

User Exec mode is the mode you log into on the switch.

It lets you perform high-level diagnostics (**show** commands, ping, traceroute etc), start sessions (Telnet, SSH), and change mode.

The default User Exec mode prompt is **awplus>**.

Privileged Exec mode

To change from User Exec to Privileged Exec mode, enter the command:

```
awplus> enable
```

Privileged Exec mode is the main mode for monitoring—for example, running **show** commands and debugging. From Privileged Exec mode, you can do all the commands from User Exec mode plus many system commands.

The default Privileged Exec mode prompt is **awplus#**.

Global Configuration mode

To change from Privileged Exec to Global Configuration mode, enter the command:

```
awplus# configure terminal
```

From Global Configuration mode, you can configure most aspects of the switch.

The default Global Configuration mode prompt is **awplus(config)#**.

Lower-level configuration modes

A number of features are configured by entering a lower-level mode from Global Configuration mode. The following table lists these features.

Table 1-1: Features configured using the lower level modes

Mode	What it configures	Command	Default prompt
Interface	Switch ports, VLANs, the management Eth port.	interface <*name*>	`awplus(config-if)#`
Class map	QoS classes, which isolate and name specific traffic flows (classes) from all other traffic.	class-map <*name*>	`awplus(config-cmap)#`
EPSR	Ethernet Protection Switching Ring, a loop protection mechanism with extremely fast convergence times.	epsr configuration	`awplus(config-epsr)#`
Line	Console port settings or virtual terminal settings for telnet.	line console 0 line vty *number*	`awplus(config-line)#`
Ping poll	Ping polling, which checks whether specified devices are reachable or not.	ping-poll <*number*>	`awplus(config-ping-poll)#`
Policy map	QoS policies, a collection of user-defined QoS classes and the default class.	policy-map <*name*>	`awplus(config-pmap)#`
Policy map class	The QoS actions to take on a class-map, and which class-maps to associate with a QoS policy. This mode is a sub-mode of Policy map mode.	(in Policy map mode) class <*name*>	`awplus(config-pmap-c)#`
MST	Multiple Spanning Tree Protocol.	spanning-tree mst configuration	`awplus(config-mst)#`
Trigger	Triggers, which run configuration scripts in response to events.	trigger <*number*>	`awplus(config-trigger)#`
VLAN database	VLANs.	vlan database	`awplus(config-vlan)#`

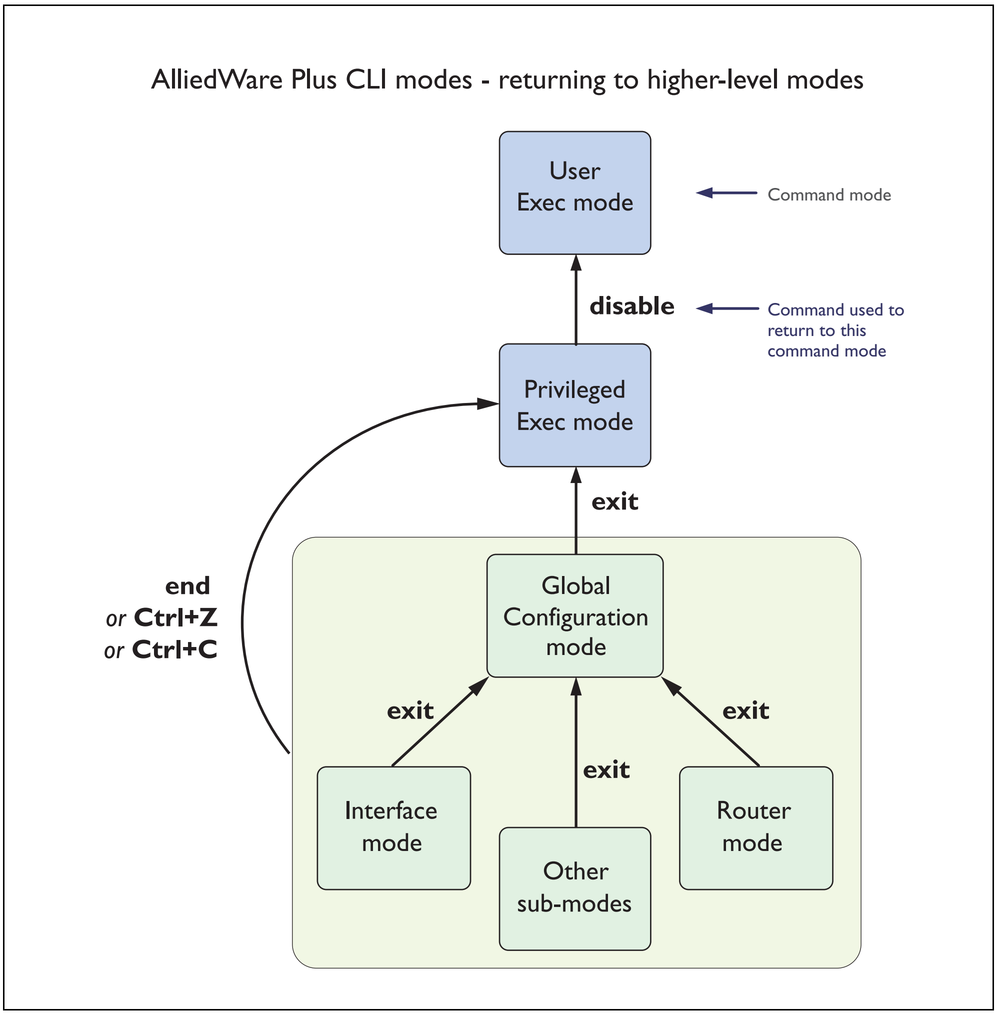
Some protocols have commands in both Global Configuration mode and lower-level configuration modes. For example, to configure MSTP, you use:

- Global Configuration mode to select MSTP as the spanning tree mode
- MST mode to create instances and specify other MSTP settings
- Interface Configuration mode to associate the instances with the appropriate ports.

Returning to higher-level modes

The following figure shows the commands to use to move from a lower-level mode to a higher-level mode.

Figure 1-9: Returning to higher-level modes



Examples To go from Interface Configuration to Global Configuration mode:

```
awplus(config-if)# exit
   awplus(config)#
```

To go from Interface Configuration to Privileged Exec mode:

```
awplus(config-if)# end
           awplus#
```

To go from Privileged Exec to User Exec:

```
awplus# exit
awplus>
```

Entering Privileged Exec Commands When in a Configuration Mode

As you configure the switch you will be constantly entering various **show** commands to confirm your configuration. This requires constantly changing between configuration modes and Privileged Exec mode.

However, you can run Privileged Exec commands without changing mode, by using the command:

```
do <command you want to run>
```

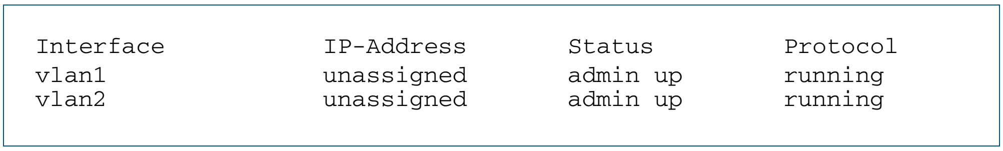
You cannot use the ? help to find out command syntax when using the do command.

Example To display information about the IP interfaces when in Global Configuration mode, enter the command:

This results in the following output:

```
awplus(config)# do show ip int brief
```

Figure 1-10: Example output after entering the command, **do show ip int brief**



```
Interface          IP-Address       Status        Protocol
vlan1              unassigned       admin up      running
vlan2              unassigned       admin up      running
```

Main Command Modes Summary

The table below lists the main command modes, how to access each mode, the prompt for each command mode. From any mode, use exit to move up a mode, or end to move to the Privileged Exec mode.

Table 1-2: Main command modes and modal prompts

Present Mode	Prompt	Command	New Mode
User Exec	`awplus>`	`enable`	Privileged Exec
Privileged Exec	`awplus#`	`configure terminal`	Global Configuration
Global Configuration	`awplus(config)#`	`vlan database`	VLAN Configuration
Global Configuration	`awplus(config)#`	`line vt <line-number>`	Line Configuration

Sub-modes Summary

The table below lists the sub-modes, how to access each mode, the prompt for each command mode, and how to exit that mode. Prompts listed use the default **awplus**.

Table 1-3: Sub-modes, prompt for each sub-mode, how to access each sub-mode, and how to exit each sub-mode

Mode	Prompt and Command Examples	How to Enter Mode	How to Exit Mode

Ping Poll Configuration	`awplus#configure terminal`
	`awplus(config)#ping-poll`
	`awplus(config-ping-poll)#`
		Use the ping-poll command available from the Global Configuration mode.	Use the exit command to return to the Global Configuration mode.
			Use the end command to return to the Privileged Exec mode.

MST (Multiple Spanning Tree) Configuration	`awplus#configure terminal`
	`awplus(config)#spanning-tree mst configuration`
	`awplus(config-mst)#`
		Use the spanning-tree mst configuration command available from the Global Configuration mode.	Use the exit command to return to the Global Configuration mode.
			Use the end command to return to the Privileged Exec mode.

Trigger Configuration	`awplus#configure terminal`
	`awplus(config)#trigger 1`
	`awplus(config-trigger)#`
		Use the trigger command from Global Configuration mode.	Use the exit command to return to the Global Configuration mode.
			Use the end command to return to the Privileged Exec mode.

EPSR Configuration	`awplus#configure terminal`
	`awplus(config)#epsr configuration`
	`awplus(config-epsr)#`
		Use the epsr configuration command available from the Global Configuration mode.	Use the exit command to return to the Global Configuration mode.
			Use the end command to return to the Privileged Exec mode.

Class Map Configuration (QoS)	`awplus#configure terminal`
	`awplus(config)#class map cmap1`
	`awplus(config-cmap)#`
		Use the class-map command available from the Global Configuration mode.	Use the exit command to return to the Global Configuration mode.
			Use the end command to return to the Privileged Exec mode.

Policy Map Configuration (QoS)	`awplus#configure terminal`
	`awplus(config)#policy-map pmap1`
	`awplus(config-pmap)#`
		Use the policy-map command available from the Global Configuration mode.	Use the exit command to return to the Global Configuration mode.
			Use the end command to return to the Privileged Exec mode.

Policy Map Class Configuration (QoS)	`awplus#configure terminal`
	`awplus(config)#policy-map pmap1`
	`awplus(config-pmap)#class cmap1`
	`awplus(config-pmap-c)#`
		Use the class command available from the Policy map mode.	Use the exit command to return to the Policy Map Configuration mode.
			Use the end command to return to the Privileged Exec mode.

How to See the Current Configuration

The current configuration is called the running-config. To see it, enter the following command in either Privileged Exec mode or any configuration mode:

```
awplus# show running-config
```

To see only part of the current configuration, enter the command:

```
awplus# show running-config|include <word>
```

This displays only the lines that contain *word*.

To start the display at a particular place, enter the command:

```
awplus# show running-config |begin <word>
```

This searches the running-config for the first instance of *word* and begins the display with that line.

Note The **show running-config** command works in all modes except User Exec mode.

Default Settings

When the switch first starts up with the AlliedWare Plus^{TM} OS, it applies default settings and copies these defaults dynamically into its running-config.

These default settings mean that the AlliedWare Plus^{TM} OS:

- encrypts passwords, such as user passwords
- records log message priority in log messages
- turns on the telnet server so that you can telnet to the switch
- enables the switch to look up domain names (but for domain name lookups to work, you have to configure a DNS server)
- turns on RSTP on all ports. Note that the ports are not set to be edge ports
- sets all the switch ports to access mode. This means they are untagged ports, suitable for connecting to hosts
- creates VLAN 1and adds all the switch ports to it
- allows logins on the serial console port
- allows logins on VTY sessions (for telnet etc)
- has switching enabled, so Layer 2 traffic is forwarded appropriately without further configuration
- has ports set to autonegotiate their speed and duplex mode
- has copper ports set to auto MDI/MDI-X mode

The Default Configuration Script

Most of the above default settings are in the form of commands, which the switch copies to its running-config when it first boots up.

The switch stores a copy of the default configuration commands in the file, **default.cfg** and uses this as its default start-up file.

For more information about start-up files, see “How to Save and Boot from the Current Configuration” on page 1.22.

The following table shows the contents of the default file.

Contents of default file	Description
`!`	An empty comment line (comments begin with an !).

`service password-encryption`
`!`
	Forces passwords in the script to be encrypted.

`log record-priority`	Records log message priority.
`username manager privilege 15 password 8 $1$bJoVec4D$JwOJGPr7YqoExA0GVasdE0`	Specifies the password for the manager user

`service telnet`
`!`
	Turns on the telnet server.

`ip domain-lookup`
`!`
	Allows domain name lookups.

`spanning-tree mode rstp`
`!`
	Turns on RSTP.

`interface port1.X.X-1.X.XX`
` switchport`
` switchport mode access`
`!`
	Sets each switch port to access mode.

`interface vlan1`
`!`
	Creates VLAN 1.

`line con 0`	A heading for any configuration settings for the console port. There are no console port settings.

`line vty 0 32`
`!`
`end`
	A heading for any configuration settings for VTY sessions. There are no VTY session settings.

How to Change the Password

To change the password for the manager account, enter Global Configuration mode and enter the following command:

```
awplus(config)# username manager password <new-password>
```

The password can be up to 23 characters in length and include characters from up to four categories. The password categories are:

- uppercase letters: A to Z
- lowercase letters: a to z
- digits: 0 to 9
- special symbols: all printable ASCII characters not included in the previous three categories.The question mark ? cannot be used as it is reserved for help functionality.

How to Set Strong Passwords

The password security rules are disabled by default. To set password security rules for users with administrative rights, or privilege level 15, enter Global Configuration mode.

You can then either specify whether the user is forced to change an expired password at the next login, or specify whether the user is not allowed to login with an expired password. You will need to specify a password lifetime greater than 0 before selecting either of these features. Note that the **security-password forced-change** and the **security-password reject-expired-pwd** commands cannot be enabled concurrently.

Password lifetime

Enter the following command to specify the password lifetime in days:

```
awplus(config)# security-password lifetime <0-1000>
```

Note that the value 0 will disable lifetime functionality and passwords will never expire. If lifetime functionality is disabled, the **security-password forced-change** command and the **security-password warning** command are also disabled.

Password forced change

To specify that a user is forced to change an expired password at the next login, enter the following command:

```
awplus(config)# security-password forced-change
```

If the **security-password forced-change** command is enabled, users with expired passwords are forced to change to a password that must comply with the current password security rules at the next login.

Reject expired password

To specify that a user is not allowed to login with an expired password, enter the following command:

```
awplus(config)# security-password reject-expired-pwd
```

If the **security-password reject-expired-pwd** command is enabled, users with expired passwords are rejected at login. Users then have to contact the Network Administrator to change their password.

Caution ⚠

Once all users' passwords are expired you are unable to login to the device again if the security-password reject-expired-pwd command has been executed. You will have to reboot the device with a default configuration file, or load an earlier software version that does not have the security password feature.

We recommend you never have the command line "security-password reject-expired-pwd" in a default config file.

Use other password security rules to further configure password security settings.

Password warning

To specify the number of days before the password expires that the user will receive a warning message specifying the remaining lifetime of the password, enter the command:

```
awplus(config)# security-password warning <0-1000>
```

The value 0 will disable warning functionality and the warning period must be less than, or equal to, the password lifetime.

Password history

To specify the number of previous passwords that are unable to be reused enter the command:

```
awplus(config)# security-password history <0-15>
```

The value 0 will disable history functionality. If history functionality is disabled, all users' password history is reset and all password history is lost. A new password is invalid if it matches a password retained in the password history.

Password minimum length

To specify the minimum allowable password length, enter the command:

```
awplus(config)# security-password minimum-length <1-23>
```

Password minimum categories

To specify the minimum number of categories that the password must contain in order to be considered valid, enter the command:

```
awplus(config)# security-password minimum-categories <1-4>
```

The password categories are:

- uppercase letters: A to Z
- lowercase letters: a to z
- digits: 0 to 9
- special symbols: all printable ASCII characters not included in the previous three categories. The question mark ? cannot be used as it is reserved for help functionality

To ensure password security, the minimum number of categories should align with the lifetime selected, i.e. the fewer categories specified the shorter the lifetime specified.

How to add a user is described in "How to Add and Remove Users" on page 1.32.

Display security password settings

To list the configuration settings for the various security password rules, enter the command:

```
awplus(config)# show security-password configuration
```

To list users remaining lifetime or last password change, enter the command:

```
awplus(config)# show security-password user
```

How to Set an IP Address on VLAN 1

This section describes how to set an IP address on the default VLAN (**vlan1**).

Step 1: If desired, check the current configuration

After logging in, enter Privileged Exec mode by using the command:

```
awplus# enable
```

Then check the current configuration by using one of the following commands:

```
awplus# show ip interface vlan1 brief
```

This results in the following output:

```
Interface              IP-Address      Status          Protocol
vlan1                  172.28.8.200    admin up        running
```

```
awplus# show running-config interface vlan1
```

This results in the following output:

```
!
interface vlan1
 ip address 172.28.8.200/16
!
```

Step 2: Enter Interface Configuration mode for the vlan1 interface

Enter Global Configuration mode and enter the command:

```
awplus(config)# interface vlan1
```

Step 3: Enter the IP address and mask

Enter the command:

```
awplus(config-if)# ip address <address/mask>
```

For example, to set the address to 172.28.8.210/16, enter the command:

```
awplus(config-if)# ip address 172.28.8.210/16
```

How to Save and Boot from the Current Configuration

This section tells you how to save your configuration and run the saved configuration when the switch starts up.

You can either:

- save the configuration to the switch's default configuration file (called "default.cfg"). By default, the switch uses that file at start-up.
- create a new configuration file and set the switch to use the new configuration file at start-up.

How to Save to the Default Configuration File

Enter Privileged Exec mode and enter the command:

```
awplus# copy running-config startup-config
```

The parameter **startup-config** is a short-cut for the current boot configuration file, which will be the default configuration file unless you have changed it, as described in the next section.

How to Create and Use a New Configuration File

Step 1: Copy the current configuration to a new file

Enter Privileged Exec mode and enter the command:

```
awplus# copy running-config <destination-url>
```

Example To save the current configuration in a file called `example.cfg`, enter the command

```
awplus# copy running-config example.cfg
```

Step 2: Set the switch to use the new file at startup

To run the new file's configuration when the switch starts up, enter Global Configuration mode and enter the command:

```
awplus(config)# boot config-file <filepath-filename>
```

Note that you can set the switch to use a configuration file on a USB flash drive if you have saved the configuration file to the drive. You can only specify that the configuration file is on a USB drive if there is a backup configuration file already specified in Flash. To set a backup configuration file to load if the main configuration file cannot be loaded, enter the command:

```
awplus(config)# boot config-file backup <filepath-filename>
```

For an explanation of the configuration fallback order, see "The configuration file fallback order" on page 6.10.

Example To run the commands in `example.cfg` on startup, enter the command:

```
awplus(config)# boot config-file flash:/example.cfg
```

To set `backup.cfg` as the backup to the main configuration file, enter the command:

```
awplus(config)# boot config-file backup flash:/backup.cfg
```

Step 3: Display the new settings

To see the files that the switch uses at startup, enter Privileged Exec mode and enter the command:

```
awplus# show boot
```

The output looks like this:

```
Boot configuration
----------------------------------------------------------------
Current software   : x510-5.4.2A.rel
Current boot image : flash:/x510-5.4.2A.rel
Backup  boot image : flash:/x510-5.4.2A.rel
Default boot config: flash:/default.cfg
Current boot config: card:/example.cfg (file exists)
Backup  boot config: flash:/backup.cfg (file exists)
```

Step 4: Continue updating the file when you change the configuration

When you next want to save the current configuration, enter Privileged Exec mode and enter the command:

```
awplus# copy running-config startup-config
```

The parameter **startup-config** is a short-cut for the current boot configuration file.

How to Return to the Factory Defaults

The switch dynamically adds the default settings to the running-config at start-up if the default file is not present. This section describes how to use this feature to return to the factory defaults.

Note After reboot the show running-config output will show the default factory settings for your switch once you have removed the default.cfg file. To recreate the default.cfg file enter copy running-config startup-config. When you enter copy running-config startup-config commands the default.cfg file is updated with the startup-config.

Completely restore defaults

To completely remove your configuration and return to the factory default configuration, delete or rename the default file and make sure no other file is set as the start-up configuration file.

To find the location of the default boot configuration file, enter Privileged Exec mode and enter the command:

```
awplus# show boot
```

To delete the default file when it is the current boot configuration file, enter Privileged Exec mode and enter either of the commands:

```
awplus# delete force <filename>
```

or:

```
awplus# erase startup-config
```

Note that erasing startup-config deletes the current boot configuration file—it does not simply stop the file from being the boot file.

To make sure that no other file is loaded at start-up, enter Global Configuration mode and enter the command:

```
awplus(config)# no boot config-file
```

Partially restore defaults

To partially restore the default settings, make a configuration file that contains the settings you want to keep and set this as the start-up configuration file. On start-up, the switch will add the missing settings to the running-config.

How to See System Information

This section describes how to view the following system information:

- overview information
- details of temperature and voltage
- serial number

Viewing Overall System Information

To display an overview of the switch hardware, software, and system settings, enter User Exec or Privileged Exec mode and enter the command:

```
awplus# show system
```

The output looks like this:

```
Switch System Status                                   Tue Aug 14 14:30:08 2012
Board       ID  Bay   Board Name                             Rev   Serial number
--------------------------------------------------------------------------------
Base       369        x510-28GTX                         X2-1   A04736H120500033
--------------------------------------------------------------------------------
RAM:  Total: 485460 kB Free: 394104 kB
Flash: 63.0MB Used: 20.1MB Available: 42.9MB
--------------------------------------------------------------------------------
Environment Status : Normal
Uptime             : 17 days 19:00:46
Bootloader version : 2.0.9-devel


Current software   : x510-5.4.2A.rel
Software version   : 5.4.2A-20120424-1
Build date         : Tue Apr 24 13:42:56 NZST 2012

Current boot config: flash:/default.cfg (file exists)
User Configured Territory: usa

System Name
 awplus
```

Viewing Temperature, Voltage, and Fan Status

The switch monitors the environmental status of the switch and its power supplies and fan. To display this information, enter User Exec or Privileged Exec mode and enter the command:

```
awplus# show system environment
```

The output looks like the following figure.

```
Stack Environment Monitoring Status


Stack member 1:


Overall Status: Normal


Resource ID: 1  Name: x510-28GTX
ID  Sensor (Units)                      Reading  Low Limit High Limit Status
1   Fan: Fan 1 (Rpm)                       4344       3000          -     Ok
2   Voltage: 1.8V (Volts)                 1.804      1.617      1.978     Ok
3   Voltage: 1.0V (Volts)                 0.995      0.896      1.099     Ok
4   Voltage: 3.3V (Volts)                 3.291      2.960      3.613     Ok
5   Voltage: 5.0V (Volts)                 5.066      4.477      5.498     Ok
6   Voltage: 1.2V (Volts)                 1.187      1.072      1.318     Ok
7   Temp: CPU (Degrees C)                    50        -10         90     Ok
```

Viewing the Serial Number

The switch's serial number is displayed in the output of the show system command on page 8.45, but for convenience, you can also display it by itself. To do this, enter User Exec or Privileged Exec mode and enter the command:

```
awplus# show system serialnumber
```

The output looks like this:

```
P1FY7502C
```

How to Set System Parameters

You can set system parameters to personalize the switch and make it easy to identify it when troubleshooting. This section describes how to configure the following system parameters:

- telnet session timeout
- switch name
- login banner

How to Change the Telnet Session Timeout

By default, telnet sessions time out after 10 minutes of idle time. If desired, you can change this.

To change the timeout for all telnet sessions, enter Global Configuration mode and enter the commands:

```
     awplus(config)# line vty 0 32
awplus(config-line)# exec-timeout <new-timeout>
```

The new timeout value only applies to new sessions, not current sessions.

Examples

To set the timeout to 30 minutes, enter the command:

```
awplus(config-line)# exec-timeout 30
```

To set the timeout to 30 seconds, enter the command:

```
awplus(config-line)# exec-timeout 0 30
```

To set the timeout to infinity, so that sessions never time out, enter either of the commands:

```
awplus(config-line)# no exec-timeout
awplus(config-line)# exec-timeout 0 0
```

How to Name the Switch

To give the switch a name, enter Global Configuration mode and enter the command:

```
awplus(config)# hostname <name>
```

For example, to name the switch "switch1.mycompany.com":

```
awplus(config)# hostname switch1.mycompany.com
```

The prompt displays the new name:

```
awplusswitch1.mycompany.com(config)#
```

The name can contain hyphens and underscore characters.

However, the name must be a single word, as the following example shows.

```
awplus(config)#hostname switch1.mycompany.com more words
hostname switch1.mycompany.com more words
                               ^

% Invalid input detected at '^' marker.
```

It also cannot be surrounded by quote marks, as the following example shows.

```
awplus(config)#hostname "switch1.mycompany.com more words"
% Please specify string starting with alphabet
```

Removing the name

To remove the hostname, enter the command:

```
awplusswitch1.mycompany.com(config)# no hostname
```

The prompt changes back to the default prompt:

```
awplus(config)#
```

How to Display a Text Banner at Login

By default, the switch displays the AlliedWare Plus^{TM} OS version and build date before login. You can customize this by changing the Message of the Day (MOTD) banner.

To enter a new MOTD banner, enter Global Configuration mode and enter the command:

```
awplus(config)# banner motd <banner-text>
```

The text can contain spaces and other printable characters. You do not have to surround words with quote marks.

Example

To display "this is a new banner" when someone logs in, enter the command:

```
awplus(config)# banner motd this is a new banner
```

This results in the following output at login:

```
awplus login: manager
Password:
this is a new banner
awplus>
```

Removing the banner

To return to the default banner (AlliedWare Plus^{TM} OS version and build date), enter the command:

```
awplus(config)# banner motd default
```

To remove the banner instead of replacing it, enter the command:

```
awplus(config)# no banner motd
```

How to Set the Time and Date

There are three aspects to setting the time and date:

- setting the current time and date ("How to Set the Time and Date" on page 1.30)
- setting the timezone ("How to Set the Timezone" on page 1.31)
- configuring the switch to automatically change the time when summer-time begins and ends ("How to Configure Summer-time" on page 1.31)

Instead of manually setting the time, you can use NTP to automatically get the time from another device.

How to Show Current Settings

To display the current time, timezone and date, enter Privileged Exec mode and enter the command:

```
awplus# show clock
```

The output looks like this:

```
UTC Time:   Wed,  3 Dec 2008 16:08:14 +0000
Timezone: UTC
Timezone Offset: +00:00
Summer time zone: None
```

How to Set the Time and Date

To set the time and date, enter Privileged Exec mode and enter the command, "clock set" on page 8.6:

```
clock set <hh:mm:ss> <day> <month> <year>
```

:where:

- *hh* is two digits giving the hours in 24-hour format (e.g. **14**)
- *mm* is two digits giving the minutes
- *ss* is two digits giving the seconds
- *day* is two digits giving the day of the month
- *month* is the first three letters of the month name (e.g. **sep**)
- *year* is four digits giving the year

Example To set the time to 14:00:00 on 25 January 2012, use the command:

```
awplus# clock set 14:00:00 25 jan 2012
```

How to Set the Timezone

To set the timezone, enter Global Configuration mode and enter the command “clock timezone” on page 8.10.

```
clock timezone <timezone-name> {plus|minus} <0-12>
```

The <*timezone-name*> can be any string up to 6 characters long.

To return the timezone to UTC+0, enter the command:

```
awplus(config)# no clock timezone
```

Example To set the timezone to Eastern Standard Time, use the command:

```
awplus(config)# clock timezone EST minus 5
```

How to Configure Summer-time

There are two approaches for setting summer-time:

- *recurring*, when you specify the week when summer-time starts and ends and each year the switch changes the time at those weeks. For example, Eastern Daylight Time (EDT) starts at 2 am on the second Sunday in March and ends at 2 am on the first Sunday in November.
- *date-based*, when you specify the start and end dates for summer-time for a particular year. For example, Eastern Daylight Time (EDT) starts at 2 am on Sunday, 8 March 2008 and ends at 2 am on Sunday, 2 November 2008.

Recurring To set summer-time with recurring dates, enter Global Configuration mode and enter the clock summer-time recurring command:

```
clock summer-time <zone-name> recurring <start-week> <start-
day> <start-month> <start-time> <end-week> <end-day>
<end-month> <end-time> <1-180>
```

The <*zone-name*> can be any string up to 6 characters long.

The <*start-time*> and <*end-time*> are in the form `hh:mm`, in 24-hour time.

Note that if you specify 5 for the week, this changes the time on the last day of the month, not the 5th week.

Example To configure EDT, enter the command:

```
awplus(config)# clock summer-time EDT recurring 2 Sun Mar 02:00
                1 Sun Nov 02:00 60
```

Date-based To set summer-time for a single year, enter Global Configuration mode and enter the clock summer-time date command:

```
clock summer-time <zone-name> date <start-day> <start-month>
<start-year> <start-time> <end-day> <end-month> <end-year>
<end-time> <1-180>
```

The <*zone-name*> can be any string up to 6 characters long.

The <*start-time*> and <*end-time*> are in the form `hh:mm`, in 24-hour time.

Example

For example, to configure EDT for 2008 enter the command:

```
awplus(config)# clock summer-time EDT date 8 Mar 2008 02:00 2
                Nov 2008 02:00 60
```

How to Add and Remove Users

Adding users

To add a new user with administrative rights, enter Global Configuration mode and enter the command:

```
awplus(config)# username <name> privilege 15 password
                <password>
```

Both <*name*> and <*password*> can contain any printable character and are case sensitive.

When you add a user with administrative rights, <*password*> will have to conform to the rules specified by the security-password minimum-categories command on page 5.17 and the security-password minimum-length command on page 5.18. If the security-password history command on page 5.14 is enabled, <*password*> is invalid if it matches a password retained in the password history.

The AlliedWare Plus^{TM} OS gives you a choice of **1** or **15** for the privilege level. Level **1** users are limited to User Exec mode so you need to set most users to level **15**.

For example, to add user Bob with password 123$%^, enter the command:

```
awplus(config)# username Bob privilege 15 password 123$%^
```

Removing users

To remove a user, enter Global Configuration mode and enter the command:

```
no username <name>
```

For example, to remove user Bob, enter the command:

```
awplus(config)# no username Bob
```

Note that you can delete all users, including the user called "manager" and the user you are logged in as. If all privilege **15** user accounts are deleted, a warning message is generated:

```
% Warning: No privileged users exist.
```

If all privilege level **15** user accounts are deleted, and there are no other users configured for the device, you may have to reboot with the default configuration file.

If there is a user account on the device with a lower privilege level and a password has already been set with the enable password command on page 5.3, you can login and still enter privileged mode. When executing the **enable** command, enter the password created with the **enable password** command. For example, if the password is mypassword:

```
awplus> enable mypassword

awplus#
```

Displaying users

To list the currently logged-in users, enter User Exec or Privileged Exec mode and enter the command:

```
awplus# show users
```

The output looks like this:

```
Line     User              Host(s)  Idle      Location         Priv Idletime Timeout
con 0    manager           idle     00:00:00  ttyS0            15   10       N/A
vty 0    bob               idle     00:00:03  172.16.11.3      1    0        5
```

To list all configured users, enter User Exec or Privileged Exec mode and enter the command:

```
awplus# show running-config |include username
```

The output looks like this:

```
username manager privilege 15 password 8 $1$bJoVec4D$JwOJGPr7YqoExA0GVasdE0
username Bob privilege 15 password 8 $1$gXJLY8dw$iqkMXLgQxbzSOutNUa5E2.
```

Pre-encrypted Passwords

The running-config output above includes the number 8 after the **password** parameter. This indicates that the password is displayed in its encrypted form.

You can enter the number 8 and a pre-encrypted password on the command line. You may want to pre-encrypt passwords if you need to load them onto switches via an insecure method (such as HTTP, or by emailing them to remote users).

Caution **Only enter the number 8 if you are entering a pre-encrypted password—otherwise, you will be unable to log in using the password and will be unable to access the switch through that username. The next section describes why.**

Testing this feature

If you want to test the effect of this, *create a new user* for the test instead of using the manager user. The test stops you from logging in as the test user, so you need to have the manager user available to log in as.

The following output shows how specifying the number 8 puts the password into the running-config exactly as you typed it:

```
awplus(config)#username Bob privilege 15 password 8 friend
awplus(config)#show running-config |include username Bob
username Bob privilege 15 password 8 friend
```

After entering the command above, logging in as “Bob” with a password of “friend” does not work. This is because the switch takes the password you enter (“friend”), hashes it, and compares the hash with the string in the running-config (“friend”). The hashed value and “friend” are not the same, so the switch rejects the login.

How to Undo Settings

There are two possibilities for undoing settings: the **no** parameter and the **default** parameter.

How to Use the *no* Parameter

To undo most settings, simply re-enter the first parameters of the configuration command with the parameter **no** before them.

Example You can set the timezone to Eastern Standard Time by entering the command:

```
awplus(config)# clock timezone EST minus 5
```

To remove the timezone setting, enter the command:

```
awplus(config)# no clock timezone
```

How to Use the *default* Parameter

Some commands have a **default** parameter that returns the feature to its default setting.

Example You can change the login banner to "this is a new banner" by entering the command:

```
awplus(config)# banner motd this is a new banner
```

To return to the default banner, enter the command:

```
awplus(config)# banner motd default
```

Note that this command also has a **no** parameter that lets you remove the banner altogether.

How to Upgrade the Firmware

New releases of the AlliedWare Plus^{TM} OS become available regularly. Contact your customer support representative for more information.

Step 1: Put the new release onto your TFTP server

Step 2: If necessary, create space in the switch's Flash memory for the new release

Note that you cannot delete the current release file.

To see how much space is free, use the command:

```
awplus# show file systems
```

Step 3: Copy the new release from your TFTP server onto the switch

Follow the instructions in "Copying with Trivial File Transfer Protocol (TFTP)" on page 6.15.

Step 4: Set the switch to boot from the new release

Enter Global Configuration mode and enter the command:

```
awplus(config)# boot system <filepath-filename>
```

You can set a backup release file to load if the main release file cannot be loaded. Enter the command:

```
awplus(config)# boot system backup <filepath-filename>
```

Step 5: Check the boot settings

Enter Privileged Exec mode and enter the command:

```
awplus# show boot
```

Step 6: Reboot

Enter Privileged Exec mode and enter the command:

```
awplus# reload
```

Save Power With the Eco-Friendly Feature

You can conserve power by enabling the eco-friendly feature with the ecofriendly led command on page 8.13. This feature disables power to the port LEDs, including the stack port status LEDs. Power to the system status and stack management LEDs are not disabled.

When the eco-friendly feature is enabled, a change of port status will not affect the display of the associated LED. When the eco-friendly feature is disabled and power is returned to port LEDs, the LEDs will correctly show the current state of the ports.

In a stack environment, enabling the eco-friendly feature on the stack master will apply the feature to every member of the stack.

The eco-friendly feature is disabled by default. To enable the feature, enter the commands:

```
        awplus# configure terminal
awplus(config)# ecofriendly led
```

To display the current eco-friendly configuration status of the switch, enter the command:

```
        awplus# show ecofriendly
```

For an example of how to configure a trigger to enable the eco-friendly feature, see “Turn Off Power to Port LEDs” on page 71.7.

Trouble-shoot fiber and pluggable issues

Diagnostics Monitoring (DDM) for SFP (1 Gigabit Small Form-factor Pluggable) and SFP+ (10 Gigabit Small Form-factor Pluggable) transceivers, allow you to measure optical parameters for pluggables installed in a switch and trouble shoot fiber issues.

Fiber cable can be vulnerable to damage. Patch panels and patch cables can be connected with the wrong type of fiber, fiber splices can become faulty and fiber cables can be cut accidentally. Trouble shooting fiber issues has required special equipment and expertise to find the source of a problem causing signal attenuation. DOM and SFP DDM features help find fiber issues.

Different types of SFP,and SFP+ pluggable transceivers are supported in different models of switch. See your Allied Telesis dealer for more information about the particular models of pluggables that your switch supports, and if these SFPs also support DDM or DOM.

To display DOM, SFP DDM or SFP+ DDM diagnostic information, depending on the model of SFP or SFP+ pluggables installed on your switch, enter the following command:

```
awplus# show system pluggable diagnostics
```

The following parameters are measured by DDM for SFP and SFP+ transceivers, and are displayed in show system pluggable diagnostics command output:

- Temperature (Centigrade) inside the SFP or SFP+ transceiver
- Vcc (Volts) voltage supplied to the SFP or SFP+ transceiver
- Tx Bias (mA) current to the Laser Diode in the SFP or SFP+ transceiver
- Tx Power (mW) the amount of light transmitted from the SFP or SFP+ transceiver
- Rx Power (mW) the amount of light received in the SFP or SFP+ transceiver

You can track Tx Bias to find out how the Laser Diode in the SFP or SFP+ transceiver is aging by comparing the Tx Bias for one SFP or SFP+ transceiver against Tx Bias for others. You can use this information to see if any SFP or SFP+ transceivers may need replacement.

You can trouble shoot fiber connectivity issues by checking the Tx Power at one end of the fiber link against the Rx Power at the other end of the fiber link to measure the attenuation. Knowing the attenuation enables you to determine if there are anomalies in the fiber cable.

Note that Tx Power differences between the same type of SFP or SFP + transceivers installed on a switch may indicate that an SFP or SFP+ transceiver is not seated or locked. Ensuring SFP or SFP+ transceivers are seated and locked in place with the retaining clip will keep the fiber link up if there is any vibration or movement that can dislodge a fiber cable. Rx Power differences may indicate poor fiber patch cables, poor connectors or poor splices. Tracking Tx Bias for installed SFP or SFP+ transceivers and measuring attenuation for fiber links allows you to perform periodic preventative maintenance, instead of reacting to a failure. Tracking Tx Power differences can be used as an indicator of failure in an SFP or SFP+ which may need replacing.

Continuous Reboot Prevention

Occasionally, due to network conditions or to recover from a software failure, the recovery mechanism of the switch is to reboot to resume normal operation. Provided the same error condition does not recur within a short period of time this is acceptable behavior. However, if the error condition repeatedly occurs within a short time period, the switch will go into a cycle of continuous reboots, causing network problems.

Although a switch continuously rebooting will come to the attention of a network administrator who can then resolve the issue, it is likely that in the meantime network problems have arisen. For example, a broadcast storm due to STP becoming unstable and trying to continually reconverge could cause the switch to reboot continuously.

In a VCStack situation, a continually rebooting switch will destabilize the stack and may cause the master and member devices to continually swap roles as they both reboot. This has a devastating effect on the network since both devices are too busy rebooting and forming the stack to forward traffic.

The continuous reboot prevention feature, enabled with the continuous-reboot-prevention command on page 8.11, allows the user to configure a switch to stop rebooting if the device gets into a cycle of continuous rebooting. The user can configure the time period, the maximum number of times the switch can reboot within the specified time period, referred to as the threshold, and the action to take if the threshold is exceeded.

There are three actions you can specify:

- linkdown

  The reboot procedure continues and all switch ports and stack ports stay link down. This is the default action.
- logonly

  The reboot procedure continues normally.
- stopreboot

  The reboot procedure stops and the user is prompted to enter the key ”c” via the CLI. Normal reboot procedure then continues.

Note that when the continuous reboot prevention feature is enabled on the switch, user initiated reboots via the CLI and software version auto-synchronization reboots (VCStack implementation) are not counted toward the threshold value.

The continuous reboot prevention feature is disabled by default. To enable the feature, enter the following commands:

```
        awplus# configure terminal

awplus(config)# continuous-reboot-prevention enable
```

Unless the **period**, **threshold** and **action** parameter values are explicitly set, the defaults are used:

- period - 600 seconds
- threshold - 1 reboot event
- action - linkdown

To configure the **period**, **threshold** and the **action** to take if the number of reboots exceeds the specified threshold, enter the following commands:

```
        awplus#  configure terminal

 awplus(config)#  continuous-reboot-prevention [period <60-
                  604800>] [threshold <1-10>]
                  [action [linkdown|logonly|stopreboot]]
```

If the action **stopreboot** is specified, the reboot procedure stops and the following message is displayed:

```
Please input key 'c' if you want to continue processing.
```

When the user has input "c" via the CLI, the reboot procedure continues.

To disable the continuous reboot prevention feature, enter the following commands:

```
        awplus#  configure terminal

 awplus(config)#  no continuous-reboot-prevention enable
```

To return either one or more of the **period**, **threshold** and the **action** parameters to the default, use the commands:

```
        awplus#  configure terminal

 awplus(config)#  no continuous-reboot-prevention [period]
                  [threshold] [action]
```

To display the current continuous reboot prevention configuration, enter the command:

```
        awplus#  show continuous-reboot-prevention
```

To display the reboot history of the switch, enter the command:

```
        awplus#  show reboot history
```

Controlling "show" Command Output

You can control the output of **show** commands by using the | and > or >> tokens in the following ways:

- To display only part of the output, follow the command with **|** and then other keywords (see **Output Modifiers** below)
- To save the output to a file, follow the command with **>** ***filename***
- To append the output to an existing file, follow the command with **>>** ***filename***

Using the ? after typing the **show** command displays the following information about these tokens:

```
awplus#  show users
```

```
| Output modifiers
> Output redirection
>> Output redirection (append)
```

Output Modifiers

Type the | (vertical bar) to use **Output modifiers**.

```
append    Append output
begin     Begin with the first line that contains matching output
exclude   Exclude lines that contain matching output
include   Include lines that contain matching output
redirect  Redirect output
```

Begin

The **begin** parameter causes the display to begin at the first line that contains the input string.

```
awplus#  show run | begin vlan1
```

```
...skipping
interface vlan1
 ip address 192.168.14.1
!!
line con 0
 login
line vty 0 4
 login
!
end
```

Exclude

The **exclude** parameter excludes all lines of output that contain the input string. In the following output all lines containing the word "input" are excluded:

```
awplus#  show interface vlan1 | exclude input
```

```
Interface vlan1
  Scope: both
  Hardware is Ethernet, address is 192.168.14.1
  index 3 metric 1 mtu 1500 <UP,BROADCAST,RUNNING,MULTICAST>
  VRF Binding: Not bound
  Label switching is disabled
  No Virtual Circuit configured
  Administrative Group(s): None
  DSTE Bandwidth Constraint Mode is MAM
    output packets 4438, bytes 394940, dropped 0
    output errors 0, aborted 0, carrier 0, fifo 0, heartbeat 0,
window 0
    collisions 0
```

Include

The include parameter includes only those lines of output that contain the input string. In the output below, all lines containing the word "input" are included:

```
awplus#  show interface vlan1 | include input
```

```
 input packets 80434552, bytes 2147483647, dropped 0, multicast
packets 0
 input errors 0, length 0, overrun 0, CRC 0, frame 0, fifo 1,
missed 0
```

Redirect

The **redirect** parameter puts the lines of output into the specified file. If the file already exists, the new output overwrites the file's contents; the new output is not appended to the existing file contents.

| redirect and **>** are synonyms.

```
awplus#  show history | redirect history.txt
```

Output Redirection

The output redirection token **>** puts the lines of output into the specified file. If the file already exists, the new output overwrites the file's contents; the new output is not appended to the existing file contents.

| redirect and **>** are synonyms.

```
awplus#  show history > history.txt
```

Append Output

The append output token **>>** adds the lines of output into the specified file. The file must already exist, for the new output to be added to the end of the file's contents; the new output is appended to the existing file contents.

| append and **>>** are synonyms.

```
awplus#  show history >> history.txt
```

Commands Available in each Mode

This appendix lists the commands available in the following command modes:

- "User Exec Mode" on page 1.43
- "Privileged Exec Mode" on page 1.44
- "Global Configuration Mode" on page 1.45

User Exec Mode

```
 awplus>   ?

Exec commands:
  clear          Reset functions
  debug          Debugging functions (see also 'undebug')
  disable        Turn off privileged mode command
  echo           Echo a string
  enable         Turn on privileged mode command
  exit           End current mode and down to previous mode
  help           Description of the interactive help system
  logout         Exit from the EXEC
  mstat          Show statistics after multiple multicast traceroutes
  mtrace         Trace multicast path from source to destination
  no             Negate a command or set its defaults
  ping           Send echo messages
  quit           Exit current mode and down to previous mode
  remote-command Remote stack member command execution
  rmon           Debugging functions (see also 'undebug')
  show           Show running system information
  ssh            Open an SSH connection
  telnet         Open a telnet connection
  terminal       Set terminal line parameters
  traceroute     Trace route to destination
```

Privileged Exec Mode

```
awplus# ?
```

```
Exec commands:
  activate          Activate a script
  cd                Change the current working directory
  clear             Reset functions
  clock             Manage clock
  configure         Enter configuration mode
  copy              Copy from one file to another
  debug             Debugging functions (see also 'undebug')
  delete            Delete a file
  dir               List the files on a filesystem
  disable           Turn off privileged mode command
  dot1x             IEEE 802.1X Port-Based Access Control
  echo              Echo a string
  edit              Text Editor
  enable            Turn on privileged mode command
  erase             Erase the system startup configuration
  exit              End current mode and down to previous mode
  help              Description of the interactive help system
  license           Activate software feature license
  logout            Exit from the EXEC
  mail              Send an email
  mkdir             Make a new directory
  move              Rename or move a file
  mstat             Show statistics after multiple multicast
                    traceroutes
  mtrace            Trace multicast path from source to destination
  no                Negate a command or set its defaults
  ping              Send echo messages
  platform          Execute built-in self-tests
  pwd               Print the current working directory
  quit              Exit current mode and down to previous mode
  reboot            Halt and perform a cold restart
  reload            Halt and perform a cold restart
  remote-command    Remote stack member command execution
  rmdir             Remove a directory
  rmon              Debugging functions (see also 'undebug')
  show              Show running system information
  ssh               Open an SSH connection
  tcpdump           Execute tcpdump
  telnet            Open a telnet connection
  terminal        Set terminal line parameters
  test            Test device functionality
  traceroute      Trace route to destination
  trigger         Automatic scripted responses to device events
  undebug         Disable debugging functions (see also 'debug')
  wait            Wait for a specified number of seconds
  write          Write running configuration to memory, file or
                 terminal
```

Global Configuration Mode

```
 awplus(config)# ?

Configure commands:
  access-list          Add an access list entry
  arp                  Address Resolution Protocol (ARP)
  auth-web-server    Web authentication server configuration
                      commands
  banner               Define a login banner
    boot                 Boot configuration
  class-map            Class map command
  clock                Manage clock
  crypto               Security Specific Commands
  cvlan                Configure C-VLAN parameters
  debug                Debugging functions (see also 'undebug')
  default              Restore default settings
  do                   To run exec commands in config mode
  dot1x                IEEE 802.1X Port-Based Access Control
  enable               Modify enable password parameters
  epsr                 Ethernet Protection Switching Ring (EPSR)
  exception            Configure exception settings
  exit                 End current mode and down to previous mode
  fib                  FIB information
  help                 Description of the interactive help system
  hostname             Set system's network name
  interface            Select an interface to configure
  ip                   Internet Protocol (IP)
ipv6                 Internet Protocol version 6 (IPv6)
  key                  Authentication key management
  lacp                 LACP commands
  line                 Configure a terminal line
  log                  Logging control
  loop-protection      Loop Protection
```

```
mac                  mac address
mail                 Send an email
max-fib-routes       Set maximum fib routes number
max-static-routes    Set maximum static routes number
maximum-access-list  Maximum access-list entries
maximum-paths        Set multipath numbers installed to FIB
mls                  Multi-Layer Switch(L2/L3)
no                   Negate a command or set its defaults
ntp                  Configure NTP
ping-poll            Ping Polling
platform             Configure global settings for the switch asic
policy-map           Policy map command
radius-server        RADIUS server configuration commands
rmon                 Remote Monitoring Protocol (RMON)
security-password    Configure strong security passwords
service              Modify use of network based services
show                 Show running system information
snmp-server          Manage snmp server
spanning-tree        Spanning tree commands
ssh                  Secure Shell
stack                Virtual Chassis Stacking (VCS)
system               System properties
telnet               Configure telnet
trigger              Select a trigger to configure
undebug              Disable debugging functions (see also'debug')
username             Establish User Name Authentication
virtual-server       Virtual-server configuration
vlan                 Configure VLAN parameters
vrrp                 VRRP configuration
```

AlliedWare Plus GUI

Information on loading and using the AlliedWare Plus^{TM} GUI is outside the scope of the main body of this reference manual. This topic is covered in a separate appendix to this document. See “Appendix C: GUI Reference”.

Chapter 2: Command Syntax Conventions in this Software Reference

The following table describes how command line interface syntax is shown in this Software Reference.

Syntax element	Example	What to enter in the command line

Keywords are shown in lowercase fixed-width font or bold variable-width font	`show spanning-tree mst`
	or
	`show ip route`
		Some keywords are required, and others are optional parameters. Type keywords exactly as they appear in the command syntax.

Number ranges are enclosed in angle-brackets < > and separated by a hyphen.	`<0-255>`	Enter a number from the range. Do not enter the angle brackets.
Placeholders are shown in lowercase italics within angle-brackets < >, or in uppercase italics	<*port-list*> or `ip dhcp pool` *NAME*	Replace the placeholder with the value you require. The placeholder may be an IP address, a text string, or some other value. See the parameter table for the command for information about the type of value to enter. Do not enter the angle-brackets.
Repeats are shown with ellipsis.	`param1...`	Enter the parameter one or more times.
Optional elements are shown in brackets: [ ]	`vlan` <*vid*> `[name` <*vlan-name*>`]`	If you need the optional parameter, enter it. Do not enter the brackets.
Required choices are enclosed in braces and separated by a vertical bar (pipe) : {|}.	`spanning-tree {mstp|rstp|stp} enable`	Enter one only of the options. Do not enter the braces or vertical bar.
Optional choices are enclosed in or brackets and separated by a vertical bar (pipe): [|]	`[param1|param2]`	If needed, enter one only of the options. Do not enter the brackets or vertical bar.
Inclusive options are enclosed in braces, and separated by brackets: {[ ] [ ]}.	`{[param1] [param2] [param3]}`	Enter one or more of the options and separate them with a space. Do not enter the braces or brackets.

Chapter 3: Start-up Sequence

AlliedWare Plus Start-up

Every switch has a start-up process. A specified version of product software must be loaded and executed. The bootloader is the executable code responsible for setting up the system and loading the release software.

The bootloader is the software that runs the unit when it first powers up, performing basic initialization and executing the product software release. As part of the start-up process of the switch, the bootloader allows you various options before running the product release software.

Previous versions of AlliedWare provide the option to boot to EPROM if a software release cannot be loaded, is unlicensed, or if selected by the user. The EPROM provides enough basic functionality to get a working software release loaded and operational on the switch. In AlliedWare Plus™ this task is handled by the bootloader.

As AlliedWare Plus™ begins its start-up process; there are two options that allow you to access either the diagnostic menu, or the bootloader menu. The following prompt is displayed when these options are temporarily available:

```
Bootloader 1.0.9 loaded
Press <Ctrl+B> for the Boot Menu
```

You can now enter one of the following two options to determine how the start-up process proceeds:

- Enter Ctrl+D to display the diagnostic menu.
- Enter Ctrl+B to display the bootloader menu.

Diagnostic Menu

Enter Ctrl+D during start-up to access the bootloader diagnostic menu, and provide options for performing various hardware tests. This can be useful as a tool for confirming a suspected hardware problem at the direction of network engineering personnel. When you enter Ctrl+D, the stage 1 diagnostics menu is displayed:

```
Bootup Stage 1 Diagnostics Menu:
  0. Restart
  1. Full RAM test
  2. Quick RAM test
  3. Battery backed RAM (NVS) test
  4. Bootloader ROM checksum test
  ----------------------------------
  7. Bootup stage 2 diagnostics menu
  ----------------------------------
  8. Quit to U-Boot shell
  9. Quit and continue booting
Enter selection ==>
```

The options in the stage 1 diagnostics menu allow you to initiate the following tests:

- RAM
  The Bootloader fully tests any/all DRAM installed in the system.
- NVS
  The Bootloader fully tests any/all non-volatile (battery backed) SRAM installed in the system.
- checksum
  The Bootloader checksum ROM memory for error detection.

For example, enter "2" to select a Quick RAM test:

```
Quick RAM test - press Q to quit, S to skip when failing
Writing pattern  ................................
Checking pattern ................................
Writing complemented pattern ....................
Checking complemented pattern ...................
Pass 1  total errors 0
```

Enter "7" to display the stage 2 diagnostics menu:

```
Entering stage 2...
Bootup Stage 2 Diagnostics Menu:
  0. Restart
  2. Test FLASH (Filesystem only)
  4. Erase FLASH (Filesystem only)
  5. Card slot test
  ----------------------------------
  8. Quit to U-Boot shell
  9. Quit and continue booting
```

The options in the stage 2 diagnostics menu allow you to initiate the following tests:

- Flash
  The Bootloader tests the user file system area of Flash. The bootloader is stored in a protected area of Flash that is not accessed by the user file system.
- Flash Erase
  The Bootloader erases the user file system area of Flash only.
- USB Card slot
  The Bootloader tests the USB slot.

Once any required tests are completed from the diagnostics menu, enter “9” to quit the diagnostic menu and continue the switch boot-up process.

Bootloader Menu

Enter Ctrl+B immediately following start-up to access the bootloader menu where boot options can be set. This chapter explains each of the boot options shown below..

```
Boot Menu:

  ------------------------------------------------------
  B. Boot backup software
  ------------------------------------------------------
  0. Restart
  1. Perform one-off boot from alternate source
  2. Change the default boot source (for advanced users)
  3. Update Bootloader
  4. Adjust the console baud rate
  5. Special boot options
  6. System information
  7. Restore Bootloader factory settings
  ------------------------------------------------------
  9. Quit and continue booting
```

Boot options

A powerful feature of AlliedWare Plus™ is the ability to boot from a variety of sources. Previously the switch was constrained to just booting off the release loaded into Flash memory. The only software release upgrade path being to load a new release into Flash memory and then set this release to be loaded at the next restart.

With AlliedWare Plus™ the switch can boot from other sources, such as a USB flash drive or a TFTP server. This provides a very flexible system, with multiple options to upgrade software releases and for system recovery.

Details of the bootloader menu options are as follows:

1. Perform one-off boot from alternate source

Enter “1” to provide the following one-off boot options:

```
Enter selection ==> 1

Select device:

  0. Return to previous menu
  ------------------------------------------------------
  1. Flash   (flash:)
  2. TFTP    (tftp://)
  4. YMODEM  (ymodem:)
  6. USB     (usb:)

Enter selection ==>
```

You can select a one-off boot from Flash, TFTP Server, YMODEM, or a USB flash drive. The selected option will be used for the next restart (only) of the switch. If you select to boot from the network, the bootloader prompts the user for the required network address details.

Note These settings are specific to the Bootloader.
They are not related in any way to what may be configured by the main software release.

When the switch is booted up using the ‘one-off’ selected source for the software release, it provides the option to copy the release just used to Flash for further/ permanent use:

```
login: manager
Password: ******
The system has been booted using the one off boot/recovery
mechanism.
Bootup has successfully completed.
Write this release to flash? (y/n):
```

2. Change the default boot source (for advanced users)

Entering “2” provides the option to set the boot source permanently.

```
NOTE: These settings are specific to the Bootloader.
  They are not related in any way to what may be configured
  by the 'boot system' command in the main software release.
Select device:

    ------------------------------------------------------
  1. Flash   (flash:)
  2. TFTP    (tftp://)
  4. YMODEM  (ymodem:)
  6. USB     (usb:)
  ------------------------------------------------------

Enter selection ==>
```

3. Update Bootloader

This option allows for the bootloader code to be updated. It is not detailed here, as it is envisioned that this would rarely need to be done, and only at the request of (and with support from) Allied Telesis engineering.

4. Adjust the console baud rate

The baud rate of the console session is set here to match the terminal program being used for management of the switch when connected directly to the asynchronous port. The switches default value is 9600. The baud rate selected can be set as the ‘new’ default for future use if preferred.

```
Select baud rate:

  0. Return to previous menu
  ------------------------------------------------------
  1. 9600
  2. 19200
  3. 38400
  4. 57600
  5. 115200
  6. 230400 (Setting can't be made permanent)
  7. 460800 (Setting can't be made permanent)

Enter selection ==>  1

Change your terminal program baud rate to 9600 and press
enter...  if for some reason you are unable to do this,
power cycle the device and the existing baud rate will be
restored.
Use this baud rate by default? (Y/N) ==> n
```

5. Special boot options

The special boot options allow for system recovery in the event of a forgotten password or to the default configuration.

```
Special boot options menu:

  0. Return to previous menu
  ------------------------------------------------------
  1. Skip startup script (Use system defaults)

Enter selection ==>
```

6. System information

The system information option provides some details on the hardware platform in use, such as CPU, memory, hardware (MAC) address and so on.

7. Restore Bootloader factory settings

This option allows the bootloader to be set back to factory defaults.

Caution **This option erases any settings that may have been configured by this menu**

Are you sure? (Y/N) ==>

The bootloader menu provides a powerful set of options for flexibility in the way software releases are upgraded on the switch, and system recovery is performed.

Start-up Sequence

The start-up sequence for a device running AlliedWare Plus™ under normal circumstances will be as seen below - this sequence will be seen when everything loads and runs as expected.

Note To enter the bootloader or diagnostic menus discussed previously, Ctrl+B or Ctrl+D must be entered when prompted before the software modules start loading.

There are three possible status results displayed for each module loaded - OK, INFO, ERROR:

- OK means that the module has loaded correctly.
- INFO means that an error occurred, but the device is usable.
- ERROR means that an error occurred and device operation may be affected.

Additional specific information accompanies an INFO or ERROR status result. For example, if a corrupt release file was set as the startup release, the following error message would be seen:

Whether an error message results in a case of the device being unusable will depend on the specific error and message, so will need to be dealt with on a case by case basis. If a software release has been corrupted, as shown on start-up, a new release may need to be loaded.

Chapter 4: CLI Navigation Commands

Command List

This chapter provides an alphabetical reference for the commands used to navigate between different modes. This chapter also provides a reference for the help and show commands used to help navigate within the CLI.

configure terminal

This command enters the Global Configuration command mode.

Syntax

```
configure terminal
```

Mode Privileged Exec

Example To enter the Global Configuration command mode (note the change in the command prompt), enter the command:

```
awplus# configure terminal
awplus(config)#
```

disable (Privileged Exec mode)

This command exits the Privileged Exec mode, returning the prompt to the User Exec mode. To end a session, use the exit command.

Syntax

```
disable
```

Mode Privileged Exec

Example To exit the Privileged Exec mode, enter the command:

```
awplus# disable
awplus>
```

Related Commands enable (Privileged Exec mode)
end
exit

do

This command lets you to run User Exec and Privileged Exec mode commands when you are in a Configuration mode.

Syntax `do <command>`

Parameter	Description
<command>	Specify the command and its parameters.

Mode Any configuration mode

Example

```
        awplus# configure terminal
awplus(config)# do ping 192.0.2.23
```

enable (Privileged Exec mode)

This command enters the Privileged Exec mode and optionally changes the privilege level for a session. If a privilege level is not specified then the maximum privilege level (15) is applied to the session. If the optional privilege level is omitted then only users with the maximum privilege level can access Privileged Exec mode without providing the password as specified by the enable password or enable secret commands. If no password is specified then only users with the maximum privilege level set with the username command can assess Privileged Exec mode.

Syntax

```
enable [<privilege-level>]
```

Parameter	Description
<privilege-level>	Specify the privilege level for a CLI session in the range <1-15>, where 15 is the maximum privilege level, 7 is the intermediate privilege level and 1 is the minimum privilege level. The privilege level for a user must match or exceed the privilege level set for the CLI session for the user to access Privileged Exec mode. Privilege level for a user is configured by username.

Mode User Exec

Usage Many commands are available from the Privileged Exec mode that configure operating parameters for the switch, so you should apply password protection to the Privileged Exec mode to prevent unauthorized use. Passwords can be encrypted but then cannot be recovered. Note that un-encrypted passwords are shown in plain text in configurations.

The username command sets the privilege level for the user. After login, users are given access to privilege level 1. Users access higher privilege levels with the enable (Privileged Exec mode) command. If the privilege level specified is higher than the users configured privilege level specified by the username command, then the user is prompted for the password for that level.

Note that a separate password can be configured for each privilege level using the enable password and the enable secret commands from the Global Configuration mode. The service password-encryption command encrypts passwords configured by the enable password and the enable secret commands, so passwords are not shown in plain text in configurations.

Example The following example shows the use of the **enable** command to enter the Privileged Exec mode (note the change in the command prompt).

```
awplus> enable
awplus#
```

The following example shows the **enable** command enabling access the Privileged Exec mode for users with a privilege level of 7 or greater. Users with a privilege level of 7 or greater do not need to enter a password to access Privileged Exec mode. Users with a privilege level 6 or less need to enter a password to access Privilege Exec mode. Use the enable password command or the enable secret commands to set the password to enable access to Privileged Exec mode.

```
awplus> enable 7

awplus#
```

Related Commands

disable (Privileged Exec mode)
enable password
enable secret
exit
service password-encryption
username

end

This command returns the prompt to the Privileged Exec command mode from any other advanced command mode.

Syntax end

Mode All command modes

Example The following example shows the use of the `end` command to return to the Privileged Exec mode directly from Interface mode.

```
          awplus# configure terminal
  awplus(config)# interface vlan2
awplus(config-if)# end
          awplus#
```

Related Commands disable (Privileged Exec mode)
enable (Privileged Exec mode)
exit

exit

This command exits the current mode, and returns the prompt to the mode at the previous level. When used in User Exec mode, the **exit** command terminates the session.

Syntax exit

Mode All command modes.

Example The following example shows the use of `exit` command to exit Interface mode, and return to Configure mode.

```
          awplus# configure terminal
  awplus(config)# interface vlan2
awplus(config-if)# exit
  awplus(config)#
```

Related Commands disable (Privileged Exec mode)
enable (Privileged Exec mode)
end

help

This command displays a description of the AlliedWare Plus^{TM} OS help system.

Syntax `help`

Mode All command modes

Example To display a description on how to use the system help, use the command:

```
awplus# help
```

Output Figure 4-1: Example output from the **help** command

```
When you need help at the command line, press '?'.

If nothing matches, the help list will be empty. Delete
characters until entering a '?' shows the available options.

Enter '?' after a complete parameter to show remaining valid
command parameters (e.g. 'show ?').

Enter '?' after part of a parameter to show parameters that
complete the typed letters (e.g. 'show ip?').
```

logout

This command exits the User Exec or Privileged Exec modes and ends the session.

Syntax `logout`

Mode User Exec and Privileged Exec

Example To exit the User Exec mode, use the command:

```
awplus# logout
```

show history

This command lists the commands entered in the current session. The history buffer is cleared automatically upon reboot.

The output lists all command line entries, including commands that returned an error.

For information on output options, see “Controlling “show” Command Output” on page 1.41.

Syntax

```
show history
```

Mode User Exec and Privileged Exec

Example To display the commands entered during the current session, use the command:

```
awplus# show history
```

Output Figure 4-2: Example output from the **show history** command

```
1 en
2 show ru
3 con t
4 route-map er deny 3
5 exit
6 ex
7 di
```

Chapter 5: User Access Commands

Introduction

This chapter provides an alphabetical reference of commands used to configure user access.

Command List

clear line console

This command resets a console line. If a terminal session exists on the line then the terminal session is terminated. If console line settings have changed then the new settings are applied.

Syntax `clear line console 0`

Mode Privileged Exec

Example To reset the console line (asyn), use the command:

```
awplus# clear line console 0

        % The new settings for console line 0 have been
        applied
```

Related Commands
clear line vty
flowcontrol hardware (asyn/console)
line
show users

clear line vty

This command resets a VTY line. If a session exists on the line then it is closed.

Syntax `clear line vty <0-32>`

Parameter	Description
<0-32>	Line number

Mode Privileged Exec

Example To reset the first vty line, use the command:

```
awplus# clear line vty 1
```

Related Commands
privilege level
line
show telnet
show users

enable password

To set a local password to control access to various privilege levels, use the enable password Global Configuration command. Use the enable password command to modify or create a password to be used, and use the no enable password command to remove the password.

Note that the enable secret command is an alias for the enable password command, and the no enable secret command is an alias for the no enable password command. Issuing a no enable password command removes a password configured with the enable secret command. The enable password command is shown in the running and startup configurations. Note that if the enable secret command is entered then enable password is shown in the configuration.

Note Do not use encrypted passwords for GUI users. The GUI requires unencrypted user passwords only - not encrypted user passwords. Do not use option 8 for GUI users.

Syntax

```
enable password [<plain>|8 <hidden>|level <1-15> 8 <hidden>]

no enable password [level <1-15>]
```

Parameter	Description
<plain>	Specifies the unencrypted password.
8	Specifies a hidden password will follow.
<hidden>	Specifies the hidden encrypted password. Use an encrypted password for better security where a password crosses the network or is stored on a TFTP server.
`level`	Privilege level <1-15>. Level for which the password applies. You can specify up to 16 privilege levels, using numbers 1 through 15. Level 1 is normal EXEC-mode user privileges for **User Exec** mode. If this argument is not specified in the command or the **no** variant of the command, the privilege level defaults to 15 (enable mode privileges) for **Privileged Exec** mode. A privilege level of 7 can be set for intermediate CLI security.

Default The privilege level for enable password is level 15 by default. Previously the default was level 1.

Mode Global Configuration

Usage This command enables the Network Administrator to set a password for entering the Privileged Exec mode when using the enable (Privileged Exec mode) command. There are three methods to enable a password. In the examples below, for each method, note that the configuration is different and the configuration file output is different, but the password string to be used to enter the Privileged Exec mode with the **enable** command is the same (**mypasswd**).

From release 5.4.1, a user can have an intermediate CLI security level set with this command for privilege level 7 to access all the show commands in Privileged Exec mode and all the commands in User Exec mode, but not any configuration commands in Privileged Exec mode.

Note that the enable password command is an alias for the enable secret command and one password per privilege level is allowed using these commands. Do not assign one password to a privilege level with enable password and another password to a privilege level with enable secret. Use enable password or enable secret commands. Do not use both on the same level.

Using Plain Passwords

The plain password is a clear text string that appears in the configuration file as configured.

```
        awplus#  configure terminal
awplus(config)#  enable password mypasswd
awplus(config)#  end
```

This results in the following show output

```
awplus#show run
Current configuration:
hostname awplus
enable password mypasswd
!
interface lo
```

Using Encrypted Passwords

Configure an encrypted password using the service password-encryption command. First, use the enable password command to specify the string that you want to use as a password (**mypasswd**). Then, use the service password-encryption command to encrypt the specified string (**mypasswd**). The advantage of using an encrypted password is that the configuration file does not show **mypasswd**, it will only show the encrypted string **fU7zHzuutY2SA.**

Note Do not use encrypted passwords for GUI users. The GUI requires unencrypted user passwords only - not encrypted user passwords. Do not use option 8 for GUI users.

```
        awplus#  configure terminal
awplus(config)#  enable password mypasswd
awplus(config)#  service password-encryption
awplus(config)#  end
```

This results in the following show output.

```
awplus#show run
Current configuration:
hostname awplus
enable password 8 fU7zHzuutY2SA
service password-encryption
!
interface lo
```

Using Hidden Passwords

Configure an encrypted password using the **HIDDEN** parameter (**8**) with the enable password command. Use this method if you already know the encrypted string corresponding to the plain text string that you want to use as a password. It is not required to use the service password-encryption command for this method. The output in the configuration file will show only the encrypted string, and not the text string

```
        awplus#  configure terminal
awplus(config)#  enable password 8 fU7zHzuutY2SA
awplus(config)#  end
```

This results in the following show output.

```
awplus#show run
Current configuration:
hostname awplus
enable password 8 fU7zHzuutY2SA
!
interface lo
```

Related Commands

enable (Privileged Exec mode)
enable secret
service password-encryption
privilege level
show privilege
username
show running-config

enable secret

To set a local password to control access to various privilege levels, use the enable secret Global Configuration command. Use the enable secret command to modify or create a password to be used, and use the no enable secret command to remove the password.

Note that the enable secret command is an alias for the enable password command, and the no enable secret command is an alias for the no enable password command. Issuing a no enable password command removes a password configured with the enable secret command. The enable password command is shown in the running and startup configurations. Note that if the enable secret command is entered then enable password is shown in the configuration.

Note Do not use encrypted passwords for GUI users. The GUI requires unencrypted user passwords only - not encrypted user passwords. Do not use option 8 for GUI users.

Syntax

```
enable secret [<plain>|8 <hidden>|level <0-15> 8 <hidden>]
```

```
no enable secret [level <1-15>]
```

Parameter	Description
<plain>	Specifies the unencrypted password.
8	Specifies a hidden password will follow.
<hidden>	Specifies the hidden encrypted password. Use an encrypted password for better security where a password crosses the network or is stored on a TFTP server.
level	Privilege level <1-15>. Level for which the password applies. You can specify up to 16 privilege levels, using numbers 1 through 15. Level 1 is normal EXEC-mode user privileges for **User Exec** mode. If this argument is not specified in the command or the **no** variant of the command, the privilege level defaults to 15 (enable mode privileges) for **Privileged Exec** mode. A privilege level of 7 can be set for intermediate CLI security.

Default The privilege level for enable secret is level 15 by default.

Mode Global Configuration

Usage This command enables the Network Administrator to set a password for entering the Privileged Exec mode when using the enable (Privileged Exec mode) command. There are three methods to enable a password. In the examples below, for each method, note that the configuration is different and the configuration file output is different, but the password string to be used to enter the Privileged Exec mode with the **enable** command is the same (**mypasswd**).

From release 5.4.1, a user can have an intermediate CLI security level set with this command for privilege level 7 to access all the show commands in Privileged Exec mode and all the commands in User Exec mode, but not any configuration commands in Privileged Exec mode.

Note that the enable secret command is an alias for the enable password command and one password per privilege level is allowed using these commands. Do not assign one password to a privilege level with enable password and another password to a privilege level with enable secret. Use enable password or enable secret commands. Do not use both on the same level.

Using Plain Passwords

The plain password is a clear text string that appears in the configuration file as configured.

```
        awplus#  configure terminal
awplus(config)#  enable secret mypasswd
awplus(config)#  end
```

This results in the following show output

```
awplus#show run
Current configuration:
hostname awplus
enable password mypasswd
!
interface lo
```

Using Encrypted Passwords

Configure an encrypted password using the service password-encryption command. First, use the enable password command to specify the string that you want to use as a password (**mypasswd**). Then, use the service password-encryption command to encrypt the specified string (**mypasswd**). The advantage of using an encrypted password is that the configuration file does not show **mypasswd**, it will only show the encrypted string **fU7zHzuutY2SA.**

Note Do not use encrypted passwords for GUI users. The GUI requires unencrypted user passwords only - not encrypted user passwords. Do not use option 8 for GUI users.

```
        awplus#  configure terminal
awplus(config)#  enable secret mypasswd
awplus(config)#  service password-encryption
awplus(config)#  end
```

This results in the following show output:

```
awplus#show run
Current configuration:
hostname awplus
enable password 8 fU7zHzuutY2SA
service password-encryption
!
interface lo
```

Using Hidden Passwords

Configure an encrypted password using the **HIDDEN** parameter (**8**) with the **enable password** command. Use this method if you already know the encrypted string corresponding to the plain text string that you want to use as a password. It is not required to use the service password-encryption command for this method. The output in the configuration file will show only the encrypted string, and not the text string:

```
        awplus#  configure terminal
awplus(config)#  enable secret 8 fU7zHzuutY2SA
awplus(config)#  end
```

This results in the following show output.

```
awplus#show run
Current configuration:
hostname awplus
enable password 8 fU7zHzuutY2SA
!
interface lo
```

Related Commands

enable (Privileged Exec mode)
enable secret
service password-encryption
privilege level
show privilege
username
show running-config

exec-timeout

This command sets the interval your device waits for user input from either a console or VTY connection. Once the timeout interval is reached, the connection is dropped. This command sets the time limit when the console or VTY connection automatically logs off after no activity.

The **no** variant of this command removes a specified timeout and resets to the default timeout (10 minutes).

Syntax

```
exec-timeout {<minutes>} [<seconds>]

no exec-timeout
```

Parameter	Description
<minutes>	<0-35791> Required integer timeout value in minutes
<seconds>	<0-2147483> Optional integer timeout value in seconds

Default The default for the **exec-timeout** command is 10 minutes and 0 seconds (**exec-timeout 10 0**)

Mode Line Configuration

Usage This command is used set the time the telnet session waits for an idle VTY session, before it times out. An **exec-timeout 0 0** setting will cause the telnet session to wait indefinitely. The command **exec-timeout 0 0** is useful while configuring a device, but reduces device security.

If no input is detected during the interval then the current connection resumes. If no connections exist then the terminal returns to an idle state and disconnects incoming sessions.

Examples To set VTY connections to timeout after 2 minutes, 30 seconds if there is no response from the user, use the following commands:

```
            awplus# configure terminal
    awplus(config)# line vty 0 32
awplus(config-line)# exec-timeout 2 30
```

To reset the console connection to the default timeout of 10 minutes 0 seconds if there is no response from the user, use the following commands:

```
            awplus# configure terminal
    awplus(config)# line console 0
awplus(config-line)# no exec-timeout
```

Validation Commands show running-config

Related Commands line
service telnet

flowcontrol hardware (asyn/console)

Use this command to enable RTS/CTS (Ready To Send/Clear To Send) hardware flow control on a terminal console line (asyn port) between the DTE (Data Terminal Equipment) and the DCE (Data Communications Equipment).

Syntax

```
flowcontrol hardware

no flowcontrol hardware
```

Mode Line Configuration

Default Hardware flow control is disabled by default.

Usage Hardware flow control makes use of the RTS and CTS control signals between the DTE and DCE where the rate of transmitted data is faster than the rate of received data. Flow control is a technique for ensuring that a transmitting entity does not overwhelm a receiving entity with data. When the buffers on the receiving device are full, a message is sent to the sending device to suspend the transmission until the data in the buffers has been processed.

Hardware flow control can be configured on terminal console lines (e.g. asyn0). For Reverse Telnet connections, hardware flow control must be configured to match on both the Access Server and the Remote Device. For terminal console sessions, hardware flow control must be configured to match on both the DTE and the DCE. Settings are saved in the running configuration. Changes are applied after reboot, clear line console, or after closing the session.

Use **show running-config** and **show startup-config** commands to view hardware flow control settings that take effect after reboot for a terminal console line.

Note that line configuration commands do not take effect immediately. Line configuration commands take effect after one of the following commands or events:

- issuing a clear line console command
- issuing a reboot command
- logging out of the current session

Examples To enable hardware flow control on terminal console line asyn0, use the commands:

```
           awplus# configure terminal
   awplus(config)# line console 0
awplus(config-line)# flowcontrol hardware
```

To disable hardware flow control on terminal console line asyn0, use the commands:

```
           awplus# configure terminal
   awplus(config)# line console 0
awplus(config-line)# no flowcontrol hardware
```

Related Commands clear line console
show running-config
speed (asyn)

length (asyn)

Use this command to specify the number of rows of output that the device will display before pausing, for the console or VTY line that you are configuring.

The **no** variant of this command restores the length of a line (terminal session) attached to a console port or to a VTY to its default length of 22 rows.

Syntax

```
length <0-512>

no length
```

Parameter	Description
<0-512>	Number of lines on screen. Specify 0 for no pausing.

Mode Line Configuration

Default The length of a terminal session is 22 rows. The **no length** command restores the default.

Usage If the output from a command is longer than the length of the line the output will be paused and the '–More–' prompt allows you to move to the next screen full of data.

A length of 0 will turn off pausing and data will be displayed to the console as long as there is data to display.

Examples To set the terminal session length on the console to 10 rows, use the command:

```
          awplus# configure terminal
   awplus(config)# line console 0
awplus(config-line)# length 10
```

To reset the terminal session length on the console to the default (22 rows), use the command:

```
          awplus# configure terminal
   awplus(config)# line console 0
awplus(config-line)# no length
```

To display output to the console continuously, use the command:

```
          awplus# configure terminal
   awplus(config)# line console 0
awplus(config-line)# length 0
```

Related Commands service terminal-length
terminal length
terminal resize

line

Use this command to enter line configuration mode for the specified VTYs or the console. The command prompt changes to show that the switch is in Line Configuration mode.

Syntax

```
line vty <first-line> [<last-line>]
```

```
line console 0
```

Parameter	Description
<first-line>	<0-32> Specify the first line number.
<last-line>	<0-32> Specify the last line number.
console	The console terminal line(s) for local access.
vty	Virtual terminal for remote console access.

Mode Global Configuration

Usage In Line Configuration mode, you can configure console and virtual terminal settings, including setting speed (asyn), length (asyn), privilege level, and authentication (login authentication) or accounting (accounting login) method lists.

To change the console (asyn) port speed, use this **line** command to enter Line Configuration mode before using the speed (asyn) command on page 8.63. Set the console speed (Baud rate) to match the transmission rate of the device connected to the console (asyn) port on your switch.

Note that line configuration commands do not take effect immediately. Line configuration commands take effect after one of the following commands or events:

- issuing a clear line console command
- issuing a reboot command
- logging out of the current session

Examples To enter Line Configuration mode in order to configure all VTYs, use the commands:

```
            awplus# configure terminal
    awplus(config)# line vty 0 32
awplus(config-line)#
```

To enter Line Configuration mode to configure the console (asyn 0) port terminal line, use the commands:

```
            awplus# configure terminal
    awplus(config)# line console 0
awplus(config-line)#
```

Related Commands

accounting login
clear line console
clear line vty
flowcontrol hardware (asyn/console)
length (asyn)
login authentication
privilege level
speed (asyn)

privilege level

This command sets a privilege level for VTY or console connections. The configured privilege level from this command overrides a specific user's initial privilege level at the console login.

Syntax

```
privilege level <1-15>
```

Mode Line Configuration

Usage You can set an intermediate CLI security level for a console user with this command by applying privilege level 7 to access all show commands in Privileged Exec and all User Exec commands. However, intermediate CLI security will not show configuration commands in Privileged Exec.

Examples To set the console connection to have the maximum privilege level, use the following commands:

```
           awplus#  configure terminal
     awplus(config)#  line console 0
awplus(config-line)#  privilege level 15
```

To set all vty connections to have the minimum privilege level, use the following commands:

```
           awplus#  configure terminal
     awplus(config)#  line vty 0 5
awplus(config-line)#  privilege level 1
```

To set all vty connections to have an intermediate CLI security level, to access all show commands, use the following commands:

```
           awplus#  configure terminal
     awplus(config)#  line vty 0 5
awplus(config-line)#  privilege level 7
```

Related Commands

enable password
line
show privilege
username

security-password history

This command specifies the number of previous passwords that are unable to be reused. A new password is invalid if it matches a password retained in the password history.

The **no security-password history** command disables the security password history functionality.

Syntax

```
security-password history <0-15>

no security-password history
```

Parameter	Description
<0-15>	The allowable range of previous passwords to match against. A value of *0* will disable the history functionality and is equivalent to the **no security-password history** command. If the history functionality is disabled, all users' password history is reset and all password history is lost.

Default The default history value is 0, which will disable the history functionality.

Mode Global Configuration

Examples To restrict reuse of the three most recent passwords, use the command:

```
        awplus# configure terminal
awplus(config)# security-password history 3
```

To allow the reuse of recent passwords, use the command:

```
        awplus# configure terminal
awplus(config)# no security-password history
```

Validation Commands

show running-config security-password
show security-password configuration

Related Commands

security-password forced-change
security-password lifetime
security-password minimum-categories
security-password minimum-length
security-password reject-expired-pwd
security-password warning

security-password forced-change

This command specifies whether or not a user is forced to change an expired password at the next login. If this feature is enabled, users whose passwords have expired are forced to change to a password that must comply with the current password security rules at the next login.

Note that to use this command, the lifetime feature must be enabled with the security-password lifetime command and the reject-expired-pwd feature must be disabled with the security-password reject-expired-pwd command.

The **no security-password forced-change** command disables the forced-change feature.

Syntax

```
security-password forced-change

no security-password forced-change
```

Default The forced-change feature is disabled by default.

Mode Global Configuration

Example To force a user to change their expired password at the next login, use the command:

```
        awplus#  configure terminal
awplus(config)#  security-password forced-change
```

Validation Commands

show running-config security-password
show security-password configuration

Related Commands

security-password history
security-password lifetime
security-password minimum-categories
security-password minimum-length
security-password reject-expired-pwd
security-password warning

security-password lifetime

This command enables password expiry by specifying a password lifetime in days.

Note that when the password lifetime feature is disabled, it also disables the security-password forced-change command and the security-password warning command.

The **no security-password lifetime** command disables the password lifetime feature.

Syntax

```
security-password lifetime <0-1000>

no security-password lifetime
```

Parameter	Description
<0-1000>	Password lifetime specified in days. A value of *0* will disable lifetime functionality and the password will never expire. This is equivalent to the **no security-password lifetime** command.

Default The default password lifetime is 0, which will disable the lifetime functionality.

Mode Global Configuration

Example To configure the password lifetime to 10 days, use the command:

```
        awplus# configure terminal
awplus(config)# security-password lifetime 10
```

Validation Commands

show running-config security-password
show security-password configuration

Related Commands

security-password history
security-password forced-change
security-password minimum-categories
security-password minimum-length
security-password reject-expired-pwd
security-password warning
show security-password user

security-password minimum-categories

This command specifies the minimum number of categories that the password must contain in order to be considered valid. The password categories are:

- uppercase letters: A to Z
- lowercase letters: a to z
- digits: 0 to 9
- special symbols: all printable ASCII characters not included in the previous three categories. The question mark ? cannot be used as it is reserved for help functionality.

Note that to ensure password security, the minimum number of categories should align with the lifetime selected, i.e. the fewer categories specified the shorter the lifetime specified.

Syntax

```
security-password minimum-categories <1-4>
```

Parameter	Description
<1-4>	Number of categories the password must satisfy, in the range *1* to *4*.

Default The default number of categories that the password must satisfy is 1.

Mode Global Configuration

Example To configure the required minimum number of character categories to be *3*, use the command:

```
        awplus# configure terminal
awplus(config)# security-password minimum-categories 3
```

Validation Commands

show running-config security-password
show security-password configuration

Related Commands

security-password history
security-password forced-change
security-password lifetime
security-password minimum-length
security-password reject-expired-pwd
security-password warning
username

security-password minimum-length

This command specifies the minimum allowable password length. This value is checked against when there is a password change or a user account is created.

Syntax

```
security-password minimum-length <1-23>
```

Parameter	Description
<1-23>	Minimum password length in the range from 1 to 23.

Default The default minimum password length is 1.

Mode Global Configuration

Example To configure the required minimum password length as 8, use the command:

```
        awplus# configure terminal
awplus(config)# security-password minimum-length 8
```

Validation Commands
show running-config security-password
show security-password configuration

Related Commands
security-password history
security-password forced-change
security-password lifetime
security-password minimum-categories
security-password reject-expired-pwd
security-password warning
username

security-password reject-expired-pwd

This command specifies whether or not a user is allowed to login with an expired password. Users with expired passwords are rejected at login if this functionality is enabled. Users then have to contact the Network Administrator to change their password.

Caution ⚠ **Once all users' passwords are expired you are unable to login to the device again if the security-password reject-expired-pwd command has been executed. You will have to reboot the device with a default configuration file, or load an earlier software version that does not have the security password feature.**

We recommend you never have the command line "security-password reject-expired-pwd" in a default config file.

Note that when the reject-expired-pwd functionality is disabled and a user logs on with an expired password, if the forced-change feature is enabled with security-password forced-change command, a user may have to change the password during login depending on the password lifetime specified by the security-password lifetime command.

The **no security-password reject-expired-pwd** command disables the reject-expired-pwd feature.

Syntax

```
security-password reject-expired-pwd

no security-password reject-expired-pwd
```

Default The reject-expired-pwd feature is disabled by default.

Mode Global Configuration

Example To configure the system to reject users with an expired password, use the command:

```
        awplus# configure terminal
awplus(config)# security-password reject-expired-pwd
```

Validation Commands

show running-config security-password
show security-password configuration

Related Commands

security-password history
security-password forced-change
security-password lifetime
security-password minimum-categories
security-password minimum-length
security-password warning
show security-password user

security-password warning

This command specifies the number of days before the password expires that the user will receive a warning message specifying the remaining lifetime of the password.

Note that the warning period cannot be set unless the lifetime feature is enabled with the security-password lifetime command.

The **no security-password warning** command disables this feature.

Syntax

```
security-password warning <0-1000>
```

```
no security-password warning
```

Parameter	Description
<0-1000>	Warning period in the range from *0* to *1000* days. A value *0* disables the warning functionality and no warning message is displayed for expiring passwords. This is equivalent to the **no security-password warning** command. The warning period must be less than, or equal to, the password lifetime set with the security-password lifetime command.

Default The default warning period is 0, which disables warning functionality.

Mode Global Configuration

Example To configure a warning period of three days, use the command:

```
        awplus#  configure terminal
awplus(config)#  security-password warning 3
```

Validation Commands

show running-config security-password
show security-password configuration

Related Commands

security-password history
security-password forced-change
security-password lifetime
security-password minimum-categories
security-password minimum-length
security-password reject-expired-pwd

service advanced-vty

This command enables the advanced-vty help feature. This allows you to use TAB completion for commands. Where multiple options are possible, the help feature displays the possible options.

The **no service advanced-vty** command disables the advanced-vty help feature.

Syntax

```
service advanced-vty
no service advanced-vty
```

Default The advanced-vty help feature is enabled by default.

Mode Global Configuration

Examples To disable the advanced-vty help feature, use the command:

```
        awplus# configure terminal
awplus(config)# no service advanced-vty
```

To re-enable the advanced-vty help feature after it has been disabled, use the following commands:

```
        awplus# configure terminal
awplus(config)# service advanced-vty
```

service http

This command enables the HTTP (Hypertext Transfer Protocol) service. The HTTP service is enabled by default and is required to support the AlliedWare Plus^{TM} GUI Java applet on a Java enabled browser. See Appendix C: GUI Reference for further information about installing and using the AlliedWare Plus^{TM} GUI.

The **no service http** command disables the HTTP feature.

Syntax

```
service http

no service http
```

Default The HTTP service is enabled by default.

Mode Global Configuration

Examples To disable the HTTP service, use the command:

```
        awplus# configure terminal
awplus(config)# no service http
```

To re-enable the HTTP service after it has been disabled, use the following commands:

```
        awplus# configure terminal
awplus(config)# service http
```

service password-encryption

Use this command to enable password encryption. This is enabled by default. When password encryption is enabled, the device displays passwords in the running config in encrypted form instead of in plain text.

Use the **no service password-encryption** command to stop the device from displaying newly-entered passwords in encrypted form. This does not change the display of existing passwords.

Note Do not use encrypted passwords for GUI users. The GUI requires unencrypted user passwords only - not encrypted user passwords.

Syntax

```
service password-encryption
```

```
no service password-encryption
```

Mode Global Configuration

Example

```
        awplus# configure terminal
awplus(config)# service password-encryption
```

Validation Commands show running-config

Related Commands enable password

service telnet

Use this command to enable the telnet server. The server is enabled by default. Enabling the telnet server starts the switch listening for incoming telnet sessions on the configured port.

The server listens on port 23, unless you have changed the port by using the privilege level command on page 5.13.

Use the **no** variant of this command to disable the telnet server. Disabling the telnet server will stop the switch listening for new incoming telnet sessions. However, existing telnet sessions will still be active.

Syntax

```
service telnet [ip|ipv6]
```

```
no service telnet |ipv6]ip|ipv6]
```

Default The IPv4|`ipv6]` telnet servers are enabled by default.

The configured telnet port is TCP port 23 by default.

Mode Global Configuration

Examples To enable|`ipv6]` the IPv4|`ipv6]` telnet servers, use the following commands:

```
        awplus# configure terminal
awplus(config)# service telnet
```

To enable the IPv6 telnet server only, use the following commands:

```
        awplus# configure terminal
awplus(config)# service telnet ipv6
```

To disable both the IPv4 and IPv6 telnet servers, use the following commands:

```
        awplus# configure terminal
awplus(config)# no service telnet
```

To disable the IPv6 telnet server only, use the following commands:

```
        awplus# configure terminal
awplus(config)# no service telnet ipv6
```

Related Commands

clear line vty
show telnet
telnet server

service terminal-length

Use this command to specify the number of rows of output that the device will display before pausing, for all console and VTY lines.

Use the **no** variant of this command to remove the length specified by this command. The default length will apply unless you have changed the length for some or all lines by using the length (asyn) command on page 5.11.

Syntax

```
service terminal-length <lines>
```

```
no service terminal-length <lines>
```

Parameter	Description
`terminal-length`	Establish system-wide terminal length configuration.
`<lines>`	`<0-512>` Number of rows that the device will display before pausing.

Mode Global Configuration

Usage This command overrides any lengths set by using the length (asyn) command on page 5.11 in Line mode.

Example To display 60 rows of text before pausing, use the following command:

```
        awplus# configure terminal
awplus(config)# service terminal-length 60
```

Related Commands

service terminal-length
terminal length
terminal resize

show security-password configuration

This command displays the configuration settings for the various security password rules.

Syntax

```
show security-password configuration
```

Mode

Privileged Exec

Example

To display the current security-password rule configuration settings, use the command:

```
awplus# show security-password configuration
```

Output

Figure 5-1: Example output from the **show security-password configuration** command

```
Security Password Configuration
Minimum password length ............................ 8
Minimum password character categories to match ..... 3
Number of previously used passwords to restrict..... 4
Password lifetime .................................. 30 day(s)
  Warning period before password expires ........... 3 day(s)
Reject expired password at login ................... Disabled
  Force changing expired password at login ......... Enabled
```

Related Commands

show running-config security-password
show security-password user

show security-password user

This command displays user account and password information for all users.

Syntax

```
show security-password user
```

Mode Privileged Exec

Example To display the system users' remaining lifetime or last password change, use the command:

```
awplus# show security-password user
```

Output Figure 5-2: Example output from the **show security-password user** command

```
User account and password information

UserName          Privilege    GUI-User   Last-PWD-Change   Remaining-lifetime
-----------------------------------------------------------------------------
manager           15           No            21 day(s) ago   9 days
alice             15           No             3 day(s) ago   27 days
bob               15           Yes           45 day(s) ago   Expired
guest              1           No            35 day(s) ago   No Expiry
```

Related Commands show running-config security-password
show security-password configuration

show privilege

This command displays the current user privilege level, which can be any privilege level in the range <1-15>. Privilege levels <1-6> allow limited user access (all User Exec commands), privilege levels <7-14> allow restricted user access (all User Exec commands plus Privileged Exec show commands). Privilege level 15 gives full user access to all Privileged Exec commands.

Syntax

```
show privilege
```

Mode User Exec and Privileged Exec

Usage From release 5.4.1, a user can have an intermediate CLI security level set with this command for privilege levels <7-14> to access all show commands in Privileged Exec mode and all commands in User Exec mode, but no configuration commands in Privileged Exec mode.

Example To show the current privilege level of the user, use the command:

```
awplus# show privilege
```

Output Figure 5-3: Example output from the **show privilege** command

```
awplus#show privilege
Current privilege level is 15
awplus#disable
awplus>show privilege
Current privilege level is 1
```

Related Commands privilege level

show telnet

This command shows the Telnet server settings.

Syntax

```
show telnet
```

Mode User Exec and Privileged Exec

Example To show the Telnet server settings, use the command:

```
awplus# show telnet
```

Output Figure 5-4: Example output from the **show telnet** command

```
Telnet Server Configuration
------------------------------------------------------------
Telnet server                : Enabled
Protocol                     : IPv4,IPv6
Port                         : 23
```

Related Commands

clear line vty
service telnet
show users
telnet server

show users

This command shows information about the users who are currently logged into the device.

Syntax `show users`

Mode User Exec and Privileged Exec

Example To show the users currently connected to the device, use the command:

```
awplus# show users
```

Output Figure 5-5: Example output from the **show users** command

```
Line     User                Host(s)  Idle      Location        Priv Idletime Timeout
con 0    manager             idle     00:00:00  ttyS0           15   10       N/A
vty 0    bob                 idle     00:00:03  172.16.11.3     1    0        5
```

Table 5-1: Parameters in the output of the **show users** command

Parameter	Description
`Line`	Console port user is connected to.
`User`	Login name of user.
`Host(s)`	Status of the host the user is connected to.
`Idle`	How long the host has been idle.
`Location`	URL location of user.
`Priv`	The privilege level in the range 1 to 15, with 15 being the highest.
`Idletime`	The time interval the device waits for user input from either a console or VTY connection.
`Timeout`	The time interval before a server is considered unreachable.

telnet

Use this command to open a telnet session to a remote device.

```
telnet {<hostname>|ip <ipv4-addr>|ipv6 <ipv6-addr>} [<port>]
```

Parameter	Description
`<hostname>`	The host name of the remote system.
`ip`	Keyword used to specify the IPv4 address or host name of a remote system.
`<ipv4-addr>`	An IPv4 address of the remote system.
`ipv6`	Keyword used to specify the IPv6 address of a remote system
`<ipv6-addr>`	Placeholder for an IPv6 address in the format `x:x::x:x`, for example, `2001:db8::8a2e:7334`
`<port>`	Specify a TCP port number (well known ports are in the range 1-1023, registered ports are 1024-49151, and private ports are 49152-65535).

Mode User Exec and Privileged Exec

Examples To connect to TCP port 2602 on the device at 10.2.2.2, use the command:

```
awplus# telnet 10.2.2.2 2602
```

To connect to the telnet server `host.example`, use the command:

```
awplus# telnet host.example
```

To connect to the telnet server `host.example` on TCP port `100`, use the command:

```
awplus# telnet host.example 100
```

To connect to the telnet server `host.example` with an IPv6 connection, use the command:

```
awplus# telnet ipv6 host.example
```

telnet server

This command enables the telnet server on the specified TCP port. If the server is already enabled then it will be restarted on the new port. Changing the port number does not affect the port used by existing sessions.

Syntax

```
telnet server {<1-65535>|default}
```

Parameter	Description
<1-65535>	The TCP port to listen on.
`default`	Use the default TCP port number 23.

Mode Global Configuration

Example To enable the telnet server on TCP port 2323, use the following commands:

```
        awplus# configure terminal
awplus(config)# telnet server 2323
```

Related Commands show telnet

terminal length

Use the **terminal length** command to specify the number of rows of output that the device will display before pausing, for the currently-active terminal only.

Use the **terminal no length** command to remove the length specified by this command. The default length will apply unless you have changed the length for some or all lines by using the length (asyn) command on page 5.11.

Syntax

```
terminal length <length>

terminal no length [<length>]
```

Parameter	Description
<length>	<0-512> Number of rows that the device will display on the currently-active terminal before pausing.

Mode User Exec and Privileged Exec

Examples The following example sets the number of lines to 15.

```
awplus# terminal length 15
```

The following example removes terminal length set previously.

```
awplus# terminal no length
```

Related Commands length (asyn)
service terminal-length
terminal resize

terminal resize

Use this command to automatically adjust the number of rows of output on the console, which the device will display before pausing, to the number of rows configured on the user's terminal.

Syntax

```
terminal resize
```

Mode User Exec and Privileged Exec

Usage When the user's terminal size is changed, then a remote session via SSH or TELNET adjusts the terminal size automatically. However, this cannot normally be done automatically for a serial or console port. This command automatically adjusts the terminal size for a serial or console port.

Examples The following example automatically adjusts the number of rows shown on the console:

```
awplus# terminal resize
```

Related Commands

length (asyn)
service terminal-length
terminal length

username

This command creates or modifies a user.

Syntax

```
username <name> privilege <0-15> password [8] <password>

username <name> privilege <0-15>

username <name> password [8] <password>

no username <name>
```

Parameter	Description	
<name>	The login name for the user. Do not use punctuation marks, such as single quotes (‘ ‘), double quotes (“ “), or colons (:) with the user login name.	
privilege	The user’s privilege level. Use the privilege levels to set the access rights for each user.	
	<0-15>	A privilege level: either 0 (no access), 1-14 (limited access) or 15 (full access). A user with privilege level 1-14 can only access higher privilege levels if an enable password has been configured for the privilege level the user attempts to access, and the user enters that password. A user at privilege level 1 can access the majority of show commands, and at privilege level 7 a user can access the majority of show commands including platform show commands. Privilege level 15 (to access the Privileged Exec command mode) is required to access configuration commands as well as show commands in Privileged Exec.
password	A password that the user must enter when logging in.	
	8	Specifies that you are entering a password as a string that has already been encrypted, instead of entering a plain-text password. The running-config displays the new password as an encrypted string even if password encryption is turned off. Note that the user enters the plain-text version of the password when logging in.
	<password>	The user’s password. The password can be up to 23 characters in length and include characters from up to four categories. The password categories are: ■ uppercase letters: A to Z ■ lowercase letters: a to z ■ digits: 0 to 9 ■ special symbols: all printable ASCII characters not included in the previous three categories. The question mark ? cannot be used as it is reserved for help functionality.

Mode Global Configuration

Usage An intermediate CLI security level (privilege level 7 to privilege level 14) allows a CLI user access to the majority of show commands, including the platform show commands that are not available at privilege level 1 to privilege level 6. Note that some show commands, such as show running-configuration and show startup-configuration, are only available at privilege level 15.

A privilege level of 0 can be set for port authentication purposes from a RADIUS server.

Examples To create the user `bob` with a privilege level of `15`, for all show commands including show running-configuration and show startup-configuration and to access configuration commands in Privileged Exec command mode, and with the password `bobs_secret`, use the commands:

```
        awplus# configure terminal
awplus(config)# username bob privilege 15 password bobs_secret
```

To create a user `junior_admin` with a privilege level of `7`, for intermediate CLI security level access to access all show commands, and the password `show_only`, use the commands:

```
        awplus# configure terminal
awplus(config)# username junior_admin privilege 7 password 
                show_only
```

Related Commands enable password
security-password minimum-categories
security-password minimum-length

Chapter 6: Creating and Managing Files

Introduction

This chapter provides information on:

- Working with files
- Creating and Using Configuration Files
- Copying Files To and From Your Device

Working With Files

The AlliedWare Plus^{TM} OS lets you create directory trees for file storage. This section shows:

- "Listing files" on page 6.2—listing files and seeing how much free space you have
- "Displaying the contents of configuration and text files" on page 6.4
- "Navigating through the filesystem" on page 6.4—identifying the current directory, changing directories, and creating and deleting directories
- "Using the editor" on page 6.6

Flash compaction

The Flash memory on the switch automatically compacts itself to recover space available from deleted files. The switch only does this when necessary, and not every file deletion causes Flash compaction. Flash compaction can occur after a file of any size is added to or deleted from the switch.

Caution

While Flash is compacting, the console is unresponsive. Do not restart the switch, as interrupting Flash compaction can damage files.

Listing files

To list files, enter Privileged Exec mode and enter the command:

```
awplus# dir
```

The output lists files and directories in order of modification date, descending. It looks like this:

```
-rw-      534 Jul 12 2011 17:52:50  stp.cfg
-rw-      534 Jul 12 2011 17:12:50  example.cfg
-rw- 12429011 Jul 12 2011 16:26:06  x510-5.4.2A.rel
```

Listing files including hidden system files

The **dir** command does not list all files—it hides system files and directories because users generally do not need to create or edit them. To list all files including system files, enter Privileged Exec mode and enter the command:

```
awplus# dir all
```

The output looks like this:

```
drwx        0 Jul 12 2011 17:16:32  ./
-rw-      401 Jul 12 2011 17:16:32  example.cfg
-rw-      534 Jul 12 2011 17:52:50  stp.cfg
-rw- 12429011 Jul 12 2011 16:26:06  x510-5.4.2A.rel
drwx      216 Jul  9 2011 11:31:18  ../
drwx        0 Jun 13 2011 04:31:51  .configs/
-rw-       17 Jun 13 2011 04:27:27  .release
drwx        0 Jul 10 2011 23:40:00  .ssh/
```

The hidden files and directories begin with a dot.

Seeing information about the filesystem

To display information about the different memory types on the switch, enter Privileged Exec mode and enter the command:

```
awplus# show file systems
```

The output includes the amount of free memory and the prefix you type to access that memory type, and looks like this:

```
 Size(b) Free(b)  Type   Flags  Prefixes   S/D/V   Lcl/Ntwk
Avail
-----------------------------------------------------------------
  31.0M    6.0M   flash    rw   flash:     static  local      Y
      -       -   system   rw   system:    virtual local      -
 499.0k  444.0k   nvs      rw   nvs:       static  local      Y
      -       -   sdcard   rw   card:      dynamic local      N
      -       -   tftp     rw   tftp:      -       network    -
      -       -   scp      rw   scp:       -       network    -
      -       -   sftp     ro   sftp:      -       network    -
      -       -   http     ro   http:      -       network    -
```

Listing files in a subdirectory

To list the contents of a directory, enter Privileged Exec mode and enter the command:

```
awplus# dir <directory-name>
```

Tip You can specify the directory with or without a / after the directory name.

Example To display the contents of a directory called "example", enter the command:

```
awplus# dir example
```

Listing files in NVS memory or on a USB flash drive

To list the contents of a directory in NVS, enter Privileged Exec mode and enter the command:

```
awplus# dir nvs:<directory-name>
```

To list the contents of a directory on a USB flash drive, enter the command:

```
awplus# dir USB:<directory-name>
```

To list the contents of a directory on a USB storage device, enter the command:

```
awplus# dir usb:<directory-name>
```

Example To display the contents of a directory in NVS called "example", enter the command:

```
awplus# dir nvs:example
```

Displaying the contents of configuration and text files

To display the contents of a file, enter Privileged Exec mode and enter the command:

```
awplus# show file <filename>
```

Example To display the contents of the file called "example.cfg", enter the command:

```
awplus# show file example.cfg
```

Navigating through the filesystem

Showing the current directory

To see which directory you are currently in, enter Privileged Exec mode and enter the command:

```
awplus# pwd
```

For the top-level directory, the output looks like this:

```
flash:/
```

Changing directories

To change to another directory, enter Privileged Exec mode and enter the command:

```
awplus# cd <directory-name>
```

To go to a directory one level higher in the directory tree, enter the command:

```
awplus# cd ..
```

Example To change to a directory called "example", enter the command:

```
awplus# cd example
```

To go up one level, which returns you to the top level directory, enter the command:

```
awplus# cd ..
```

Changing to a directory in NVS memory or a USB flash drive

To change to the top-level directory in the NVS memory filesystem, enter Privileged Exec mode and enter the command:

```
awplus# cd nvs:
```

To change to the top-level directory on a USB flash drive, enter the command:

```
awplus# cd USB:/
```

Next, you can change to other directories by entering the command:

```
awplus# cd <directory-name>
```

Alternatively, you can go straight from Flash to a subdirectory in the alternative filesystem, by entering one of the commands:

```
awplus# cd nvs:<directory-name>
```

```
awplus# cd USB:/<directory-name>
```

To return to the Flash filesystem, enter the command:

```
awplus# cd flash:/
```

Example To change to the directory within NVS called “example”, enter the command:

```
awplus# cd nvs:example
```

To go up one level, which returns you to the top-level directory of NVS memory, enter the command:

```
awplus# cd ..
```

Creating new directories

To create a directory, enter Privileged Exec mode and enter the command:

```
awplus# mkdir <directory-name>
```

Example To make a directory called “example” within the Flash filesystem, enter the command:

```
awplus# mkdir example
```

Deleting directories

To delete an empty directory, enter Privileged Exec mode and enter the command:

```
awplus# rmdir <directory-name>
```

To delete a directory and all its contents, enter Privileged Exec mode and enter the command:

```
awplus# delete recursive <directory-name>
```

The switch prompts you for confirmation.

Example To delete an empty directory called “example” from within the Flash filesystem, enter the command:

```
awplus# rmdir example
```

Using the editor

The inbuilt editor is JOE (Joe’s Own Editor).

To edit an existing file, enter Privileged Exec mode and enter the command:

```
awplus# edit <filename>
```

To open the editor with an empty file, enter the command:

```
awplus# edit
```

When you save the new file, you may need to specify the filesystem to store it on. For Flash, use **flash:/<filename>**.

Using JOE To format and manipulate text in JOE, you use control-character sequences. The following table summarizes a few useful sequences—for details, see: joe-editor.sourceforge.net/manpage.html.

Function	Control-character sequence
Access the help	Ctrl-K-H
Save the file without exiting (for new files, this prompts for a filename)	Ctrl-K-D
Save the file and exit (this prompts for a filename)	Ctrl-K-X
Exit without saving the file	Ctrl-C
Go to the beginning of the file	Ctrl-K-U
Go to the end of the file	Ctrl-K-V
Go up one full screen of text in the file	Ctrl-U
Go down one full screen of text in the file	Ctrl-V
Select a block of text:	
Mark the beginning of the block	Ctrl-K-B

Function	Control-character sequence
Mark the end of the block	Ctrl-K-K
Copy and paste a selected block of text	Place cursor at destination then enter Ctrl-K-C
Move a selected block of text	Place cursor at destination then enter Ctrl-K-M
Delete a selected block of text	Ctrl-K-Y

Creating and Using Configuration Files

This section provides instructions on:

- Creating a configuration file
- Specifying the start-up configuration script
- Working with configuration files

Creating a configuration file

A **configuration file** is a text file that contains a sequence of standard commands for a specific purpose. Configuration files have a **.cfg** extension. Your device has a default configuration script called **default.cfg**.

You can create and edit configuration files on your device by:

- saving the dynamic configuration on the device, known as the **running-config** (see "Working with configuration files"). Use the command:

  ```
  awplus# copy running-config (destination-URL)
  ```

  Where URL specifies a file in Flash.

- using the device's text editor. Use the command:

  ```
  awplus# edit (source-URL)
  ```

  where **source-URL** is the name of the copied file in Flash memory.

- creating a file on a remote PC, then copying it to onto your device. See "Copying files" for more information about using the **copy** commands.

Once you have created a configuration file, you can use it as the **startup-config** file. See "Specifying the start-up configuration script" for more information.

Specifying the start-up configuration script

When you restart your device, or when it automatically restarts, it executes the pre-configured commands in a configuration script known as the **boot config** or **startup-config** file.

When you first start your device, the script set as the startup-config file is **default.cfg**. If desired, you can overwrite **default.cfg** with another configuration. Alternatively, you can change the startup-config by specifying a new file as the startup-config. Use the command:

```
awplus(config)# boot config-file URL
```

where **URL** specifies the name and location of a configuration file. At the next restart, the device executes the commands in the specified file.

You can specify that the configuration file is either in the Flash or the USB flash drive. However, if you specify that the configuration file is on a USB flash drive then you must first create a backup configuration file stored in Flash. To specify a backup configuration file, use the command:

```
awplus(config)# boot config-file backup URL
```

where **URL** specifies the name and location of a configuration file.

You can change the content of the file set as the startup-config file by:

- entering commands directly into the CLI, then saving this configuration using the command:

  ```
  awplus# copy running-config startup-config
  ```

  This command saves the device's dynamic configuration into the file that is currently configured as the startup-config file.
- writing commands into a configuration file (see "Creating a configuration file" below), then using the command:

  ```
  awplus# copy SOURCE-URL startup-config
  ```

  This command saves the script from the source file into the file that is currently configured as the startup-config file.

To display the name of the configuration file that is set to execute when the device restarts, enter the command:

```
awplus# show boot
```

To see the commands in the startup-config file, use the command:

```
awplus# show startup-config
```

To erase the file set as the startup-config file, use the command:

```
awplus# erase startup-config
```

At the next restart that occurs after you've erased the file, the device loads the configuration in the file **default.cfg**. This file is set on the system as a backup configuration file that loads if no other file is set as the startup-config file.

Working with configuration files

When you use the CLI or GUI to configure your device, it stores this dynamic configuration as a list of commands called the **running-config**. To view the device's running-config, use the command:

```
awplus# show running-config
```

If you turn off the device or restart it, any unsaved changes to the running-config are lost. To save the running-config as a configuration script, use the command:

```
awplus# copy running-config destination-url
```

You may have many configuration files. Storing them on a device allows you to keep a backup device with configuration scripts for every device in the network to speed up network recovery time. Multiple scripts also let you test new configuration scripts before setting them as the startup-config. For example, to test a new script named test.cfg, enter the command:

```
awplus# copy flash:/test.cfg running-config
```

This allows you to run a configuration file any time without restarting the device, by replacing the system's current dynamic configuration with the script in the configuration file. However, note that some commands require you to restart the device before they can take effect, such as the **platform** commands.

You can also set a trigger to automatically execute a configuration script when a predetermined event occurs. For information about creating triggers, see Chapter 70, Triggers Introduction.

The configuration file fallback order

The configuration fallback order is: configuration file, backup configuration file, default configuration file and then the factory default configuration. It is important to note the there is a distinction in system behavior between when writing to the startup-config file and when the system boots up.

When you copy a configuration script from a source file into the startup-config file the system will write to the first file that is configured. Potentially, this means that if a configuration file and a backup configuration file are not set you will write to the default.cfg.

At system startup the device goes through the fallback sequence until it finds a file that exists. For example, if the configuration file is not found then the backup configuration file becomes the current boot configuration, or startup-config, and so on. In the output displayed by the **show boot** command, the **Current boot config** parameter shows the startup-config file that the switch will load during the next boot cycle. The fallback sequence when configuration files are deleted is shown below in output from the **show boot** command.

In the example output below, the current boot configuration file, **my.cfg**, is set on the USB flash drive. This is the startup-config file that the device loads at the next boot cycle.

```
awplus#show boot
Boot configuration
----------------------------------------------------------------
Current software   : x510-5.4.2A.rel
Current boot image : USB:/x510-5.4.2A.rel
Backup  boot image : flash:/x510-5.4.2A.rel
Default boot config: flash:/default.cfg
Current boot config: card:/my.cfg (file exists)
Backup  boot config: flash:/backup.cfg (file exists)
```

In the example output below, the **no boot-config** command has been used to delete the configuration file **my.cfg** onthe USB flash drive. The backup configuration file **backup.cfg** in Flash then becomes the current boot config.

```
awplus#show boot
Boot configuration
----------------------------------------------------------------
Current software   : x510-5.4.2A.rel
Current boot image : USB:/x510-5.4.2A.rel
Backup  boot image : flash:/x510-5.4.2A.rel
Default boot config: flash:/default.cfg
Current boot config: flash:/backup.cfg (file exists)
Backup  boot config: flash:/backup.cfg (file exists)
```

In the example output below, the **no boot-config backup** command has been used to delete the backup configuration file **backup.cfg**. The default configuration file **default.cfg** then becomes the current boot config.

```
awplus#show boot
Boot configuration
----------------------------------------------------------------
Current software   : x510-5.4.2A.rell
Current boot image : USB:/x510-5.4.2A.rel
Backup  boot image : flash:/x510-5.4.2A.rel
Default boot config: flash:/default.cfg
Current boot config: flash:/default.cfg (file exists)
Backup  boot config: Not set
```

If the current boot configuration file is set on a USB flash drive and then this card has been removed from the switch, the **Current boot config** parameter field indicates that this file cannot be found, as shown in the following example output.

```
awplus#show boot
Boot configuration
----------------------------------------------------------------
Current software   : x510-5.4.2A.rel
Current boot image : USB:/x510-5.4.2A.rel
Backup  boot image : flash:/x510-5.4.2A.rel
Default boot config: flash:/default.cfg
Current boot config: card:/my.cfg (file not found)
Backup  boot config: flash:/backup.cfg (file exists)
```

At system startup the switch will load the backup configuration file as the startup-config.

Copying Files To and From Your Device

This section provides instructions on:

- URL syntax
- Copying files

URL syntax

Many of the file management commands use the placeholder "URL" to represent the name and location of the file that you want to act on. The following table explains the syntax of this URL for each different type of file location.

When you copy a file...	Use this syntax:
In local Flash memory	`flash:[/][DIRECTORY/]FILENAME`
Stored on a USB storage device	`usb[:][/][DIRECTORY/]FILENAME`
Copying with Hypertext Transfer Protocol (HTTP)	`http://[[USERNAME:PASSWORD]@]{HOSTNAME | HOST-IP}[/FILEPATH]/FILENAME`
Copying with Trivial File Transfer Protocol (TFTP)	`tftp:[[//LOCATION]/DIRECTORY]/FILENAME`
Copying with Secure Copy (SCP)	`scp://USERNAME@LOCATION[/DIRECTORY][/FILENAME]`
Copying with SSH File Transfer Protocol (SFTP)	`sftp:[[//LOCATION]/DIRECTORY]/FILENAME`

Copying files

To copy files, use the **copy** commands. These commands allow you to copy files:

- between different memory types attached to your device. Use the command:

  ```
  awplus# copy <local-source local-dest filename>
  ```

  See "Copying within a filesystem" and "" for further details.

- across a serial connection using ZMODEM. Use the command:

  ```
  awplus# copy zmodem
  ```

  See "Copying with ZMODEM" for further details.

- from your device onto a remote device, or to your device from a remote device. To copy a file across an interface with IP configured, use the command:

  ```
  awplus# copy <source-url destination-url>
  ```

  To copy files across these interfaces you can use the following protocols:

  « “Copying with Hypertext Transfer Protocol (HTTP)”

  « “Copying with Trivial File Transfer Protocol (TFTP)”

  « “Copying with Secure Copy (SCP)”

  « “Copying with SSH File Transfer Protocol (SFTP)”

Copying within a filesystem

Within a directory

To copy a file within the same directory, enter Privileged Exec mode and enter the command:

```
awplus# copy <source-filename> <destination-filename>
```

If the file already exists, the switch asks whether to overwrite it, with a message like this:

```
Overwrite flash:/example.cfg? (y/n)[n]:
```

To overwrite, press the “y” key then the Enter key.

Between directories

To copy a file to another directory within the same filesystem, enter the command:

```
awplus# copy <source-filename> <directory-name>
```

The / after the directory name is required. Otherwise the switch displays an error (“37: Destination file is a directory”).

The switch then prompts you for the destination filename. To give the copy a new name, type the name at the prompt. You can include directory names in the path.

To use the same filename as the original, press the Enter key (do not press the “y” key—that names the copy “y”).

Example

To put a copy of example.cfg into the example directory, enter the command:

```
awplus# copy example.cfg example/
```

The prompt and messages look like this:

```
Enter destination file name [example.cfg]:
Copying from source file, please wait...
Copying to destination file, please wait...
0: Successful operation
```

Copying to and from NVS or or USB card

To copy between filesystems, you need to specify the filesystem prefix (`nvs:,usb:`).

For example, to copy from Flash to NVS when your current directory is the top-level Flash directory, enter Privileged Exec mode and enter the command:

```
awplus# copy <source-filename> nvs:
```

The switch prompts you for the filename, as described in the previous section.

To copy from NVS to Flash when your current directory is the top-level Flash directory, enter the command:

```
awplus# copy nvs:<source-filename> <destination-filename>
```

Example To copy the file "example.txt" from the directory in NVS called "example" to the top level of Flash, enter the command:

```
awplus# copy nvs:example/example.txt example.txt
```

Copying with ZMODEM

ZMODEM allows you to copy files from a network host over an asynchronous port. Use the command:

```
awplus# copy zmodem
```

to open Minicom and transfer a file. Alternatively you can specify the file name within the command:

```
awplus# copy SOURCE-URL zmodem
```

For example, to copy the file "july.cfg" from Flash memory using ZMODEM, use the command:

```
awplus# copy flash:/july.cfg zmodem
```

Copying with Hypertext Transfer Protocol (HTTP)

You device has a built-in HTTP client. The HTTP client enables the device to act as a browser by sending HTTP "get" or "post" requests to an HTTP server. The client is enabled by default.

For example, to load the file "bob.key" onto Flash from the security directory on the web server at www.company.com, use the command:

```
awplus# copy http://www.company.com/security/bob.key
        flash:/bob.key
```

Copying with Trivial File Transfer Protocol (TFTP)

TFTP runs over User Datagram Protocol (UDP). It is simpler and faster than FTP but has minimal capability, such as no provisions for user authentication.

To copy a file from a TFTP server to Flash memory, enter Privileged Exec mode and enter the command:

```
awplus# copy tftp flash
```

Note You can specify the server and filename in the command instead of waiting for prompts. Use a format like the following:

```
copy tftp://172.1.1.1/example.cfg flash
```

The switch prompts you for the:

- TFTP server hostname (you can enter its IP address instead)
- source filename on the TFTP server
- destination filename in Flash on the switch

To copy a file from Flash to a TFTP server, enter the command:

```
awplus# copy flash tftp
```

Follow the prompts for source filename, server, and destination filename.

If the file is not in the top level of the TFTP server, include the path as part of the filename.

Example To copy example.cfg to the TFTP server at **172.1.1.1**, enter the command:

```
awplus# copy flash tftp
```

The prompts, responses, and messages look like this:

```
Enter source file name []:example.cfg
Enter destination host name []:172.1.1.1
Enter destination file name [example.cfg]:
Copying from source file, please wait...
Copying to destination file, please wait...
0: Successful operation
```

To load the file "bob.key" from a TFTP server, where the file is in the folder "security", use the command:

```
awplus# copy tftp://security/bob.key flash:/bob.key
```

Copying with Secure Copy (SCP)

Secure Copy (SCP) provides a secure way to copy files to and from a remote device using SSH. The AlliedWare Plus^{TM} OS includes both a SSH server and a SSH client. You must enable the SSH server before your device accepts connections from SCP clients. See the Chapter 49, Secure Shell (SSH) Introduction for more information.

For example, to load the file "beth.key" onto Flash from the key directory on a remote SSH server at 10.10.0.12, using the username "bob", use the command:

```
awplus# copy scp://bob@10.10.0.12/key/beth.key
        flash:/beth.key
```

Copying with SSH File Transfer Protocol (SFTP)

SSH File Transfer Protocol (SFTP) provides a secure way to copy files onto your device from a remote device. The AlliedWare Plus^{TM} OS includes both a SSH server and a SSH client. SFTP provides additional features from SCP, such as allowing you to manipulate the remote files, and halt or resume file transfers without closing the session.

For example, to load the file "rei.cfg" onto Flash memory from the remote server at 10.0.0.5, use the command:

```
awplus# copy sftp://10.0.0.5/rei.cfg flash:/rei.cfg
```

Copying from a Server to Running Configuration

Use the **copy tftp** variant of the copy running-config command on page 7.13 to load a configuration file from a server to the running configuration of the switch.

The configuration will be added to the running configuration as if the commands were typed in the command line interface.

The resulting configuration file will be a combination of the previous running configuration and the loaded configuration file. The loaded configuration file has precedence.

The Autoboot Feature

The Autoboot feature enables your switch to automatically load a specific release file and/or configuration file from external media, such as USB Flash drive, into Flash memory, providing there is enough free space available. If there is not enough free space, the Autoboot feature will exit and booting will revert to what was previously set by the CLI. This feature is enabled only the first time the device is powered up in the field. Subsequently, the Autoboot feature is disabled by default.

The Autoboot feature minimizes network downtime by avoiding the need for manual configuration of a replacement device.

If you use prepared external media for the first time boot, the Autoboot feature gives you the ability to easily ensure the device boots with your desired release and configuration files. You must prepare the external media for this purpose using an initiation file, `autoboot.txt`, and accompanying release and configuration files.

Use the create autoboot command to create an `autoboot.txt` file on external media. This command will automatically ensure that the keys and values that are expected in this file are correct. After the file is created the command will copy the current release and configuration files across to the external media. The `autoboot.txt` file is read/writable by any desktop operating system currently supported by the AlliedWare Plus™ Operating System. Note that the external media file system is not case sensitive.

When the Autoboot feature is enabled, the device on boot-up:

- checks for a special file called `autoboot.txt` on external media, and if this file exists,
- checks in the file for the "key=value" pair "`Copy_from_external_media_enabled=yes`", and if this enable flag is set,
- loads the release file and/or configuration file from external media.

An example of a valid `autoboot.txt` file is shown in Figure 6-1 below.

Figure 6-1: Example **autoboot.txt** file

```
; J Smith,  x510-28GTX,  14 Aug 2012
[AlliedWare Plus]
Copy_from_external_media_enabled=yes
Boot_Release=x510-5.4.2A.rel
Boot_Config=network1.cfg
```

If external media is not present, cannot be read, or the internal enable flag is not set to **yes** in the switch, the switch will boot as normal. Incompatible release files are prevented from loading onto the switch, even if the enable flag is set on the switch. If there is an incompatible release file then the configuration file referenced in the `autoboot.txt` file is also not loaded onto the switch.

We recommend that no directories are present on external media used to hold the `autoboot.txt` file. In addition, large numbers of files on external media may slow the booting process.

Note Do not remove external media part way through the copy process as this may leave the device in an unstable state.

Configuration files placed on external media reduce security. Therefore, ensure adequate security precautions are taken with external media holding configuration files.

Configuration commands that rely on the presence of a feature license will fail when executed in the replacement switch if the replacement switch does not have the same feature license present.

The bootloader version on the device must be 1.1.6 or greater to support external media. An `autoboot.txt` file on a USB flash drive will not be detected on a device with a bootloader version less than 1.1.6.

Restoring a switch using Autoboot from external media

The example below describes the sequence of events when a switch in the field fails and is restored using this feature:

1. Using the **create autoboot** command, a network engineer has previously manually created a restore external media device, such as a USB flash drive. The external media device contains the following components:

   « An `autoboot.txt` file with required contents

   « An appropriate release file

   « A configuration file

2. A switch fails in the field.

3. A replacement switch of same model is installed.

Note This replacement switch must already have a xx541-2.6 or later release or bootloader version 1.1.6 or later pre-installed to be able to detect and interpret the `autoboot.txt` file.

4. The previously created external media device is placed into the replacement switch.

5. The switch powers up using its pre-installed release if present. It automatically checks the external media device for the `autoboot.txt` file.

6. The switch finds a valid `autoboot.txt` file on the external media device, with the value "Copy_from_external_media_enabled" set. The release file and configuration file both exist on the external media device.

7. The MD5sum of pre-installed Flash release file is compared to the MD5sum of the release file stored in the external media device. If they do not match, because the release file in the replacement switch is either missing or different, then the release is restored from the external media device. If the release files already match, then the release file is not copied from the external media device.

8. The MD5sum of the Flash configuration file `default.cfg` (if pre-installed in the replacement switch) is compared to the MD5sum of the configuration file stored in the external media device. If they do not match, because the configuration file in the replacement switch is either missing or different, then the configuration file is restored from the external media device. If the configuration files already match, then the configuration file is not copied from the external media device.

9. The memory space available in the switch Flash is checked to ensure the release and configuration files stored in the external media device will fit. If there is not enough space the Autoboot feature will exit.

10. The release file and configuration files are automatically copied from the external media device to switch Flash memory. The switch release and configuration files are updated to contain the appropriate names.

11. The switch is automatically rebooted.

12. The replacement switch is now running the restored release and configuration files. Subsequent reboots are based on the restored release and configuration files stored in the switch Flash memory.

13. If you want to Autoboot from external media on this specific switch in the future, you must now manually enable the Autoboot feature in the configuration menu via the **autoboot enable** command. This command resets the enable flag stored internally in the switch NVS memory.

Configure Autoboot

This section describes the commands used to configure the Autoboot feature.

Chapter 7: File Management Commands

Introduction

This chapter provides an alphabetical reference of AlliedWare Plus™ OS file management commands.

URL Syntax and Keyword Usage

Many of the commands in this chapter use the placeholder "URL" to represent the name and location of the file that you want to act on. The following table explains the syntax of this URL for each different type of file location.

When you copy a file...	Use this syntax:

In local Flash memory	`[DIRECTORY/]FILENAME`
	or
	`flash[:][/][DIRECTORY/]FILENAME`

Stored on a USB storage device	`usb[:][/][DIRECTORY/]FILENAME`
Using Hypertext Transfer Protocol (HTTP)	`http[://][[USERNAME:PASSWORD]@]{HOSTNAME | HOST-IP}[/FILEPATH]/FILENAME`
Using Trivial File Transfer Protocol (TFTP)	`tftp[:][[//LOCATION]/DIRECTORY]/FILENAME`
Using Secure Copy (SCP)	`scp[://]USERNAME@LOCATION[/DIRECTORY][/FILENAME]`
Using SSH File Transfer Protocol (SFTP)	`sftp[:][[//LOCATION]/DIRECTORY]/FILENAME`

Stored on stack member Flash to stack master Flash	`<stack_hostname>-<stack_member_id>/flash:[/][DIRECTORY]`
	`<stack_member_filename> <stack_master_filename>`

Note When the Flash base directory is required for local filesystems you may use **flash** or **flash:** or **flash:/**. Similarly, when the USB storage device base directory is required you may use any of the following terms: **usb** or **usb:** or **usb:/**.

The keywords **flash**, **nvs**, **usb**, **tftp**, **scp**, **sftp** and **http** are reserved for tab completion when using the **copy**, **move**, **delete**, **cd**, and **dir** commands.

Keywords **flash**, **nvs**, **usb**, **tftp**, **scp**, **sftp** and **http** cannot be applied as directory or subdirectory names when using a **mkdir** command.

A leading slash (/) indicates the root of the current filesystem location.

Command List

autoboot enable

This command enables the device to restore a release file and/or a configuration file from external media, such as a USB storage device. When the Autoboot feature is enabled, the device looks for a special file called `autoboot.txt` on the external media. If this file exists, the device will check the key and values in the file and recover the device with a new release file and/or configuration file from the external media. An example of a valid `autoboot.txt` file is shown in Figure 7-1 below.

Figure 7-1: Example autoboot.txt file

```
; J Smith,  x510-28GTX,  14 Aug 2012
[AlliedWare Plus]
Copy_from_external_media_enabled=yes
Boot_Release=x510-5.4.2A.rel
Boot_Config=network1.cfg
```

Use the **no** variant of this command to disable the Autoboot feature.

Note This command is not supported in a stacked configuration.

Syntax

```
autoboot enable
no autoboot enable
```

Default

The Autoboot feature operates the first time the device is powered up in the field, after which the feature is disabled by default.

Mode

Global Configuration

Example

To enable the Autoboot feature, use the command:

```
        awplus# configure terminal
awplus(config)# autoboot enable
```

To disable the Autoboot feature, use the command:

```
        awplus# configure terminal
awplus(config)# no autoboot enable
```

Related Commands

create autoboot
show autoboot
show boot

boot config-file

Use this command to either set the configuration file to use during the next boot cycle, or to set a backup configuration file to use if the main configuration file cannot be accessed.

Use the **no** variant of this command to delete either the configuration file or the backup configuration file.

Syntax

```
boot config-file [<filepath-filename>|backup <filepath-filename>]
```

```
no boot config-file [backup]
```

Parameter	Description
<filepath-filename>	Filepath and name of a configuration file. The specified configuration file must exist in the Flash or USB storage device filesystem. Backup configuration files must be in the Flash filesystem. Valid configuration files must have a **.cfg** extension.
`backup`	The specified file is a backup configuration file.

Mode Global Configuration

Usage You can only specify that the configuration file is on a USB storage device if there is a backupconfiguration file already specified in Flash. If you attempt to set the configuration file on a USB storage device and a backup configuration file is not specified in Flash, the following error message is displayed:

```
% Backup configuration files must be stored in the flash
filesystem
```

In a VCStack configuration you can only specify that the configuration file is on a USB storage device if there is a USB device inserted in all stack members. If a stack member has a USB device removed, an error message is displayed.

In addition, you can only specify that the configuration file is on a USB device if the device is writable. For example, if you attempt to set the configuration file on a USB device and stack member 2 has a write protected device inserted, an error message is displayed:

Examples To run the configuration file branch.cfg stored on the switch's Flash filesystem the next time the device boots up, use the commands:

```
        awplus# configure terminal
awplus(config)# boot config-file flash:/branch.cfg
```

To set the configuration file backup.cfg as the backup to the main configuration file, use the commands:

```
        awplus# configure terminal
awplus(config)# boot config-file backup flash:/backup.cfg
```

To run the configuration file branch.cfg stored on the switch's USB storage device filesystem the next time the device boots up, use the commands:

```
        awplus# configure terminal
awplus(config)# boot config-file usb:/branch.cfg
```

Related Commands boot system
show boot

boot system

Use this command to either set the release file to load during the next boot cycle, or to set a backup release file to load if the main release file cannot be loaded.

Use the no variant of this command to delete either the release file or the backup release file.

Syntax boot system [<filepath-filename>|backup <filepath-filename>]

no boot system [backup]

Parameter	Description
<filepath-filename>	Filepath and name of a release file. The specified release file must exist and must be stored in the root directory of the Flash or USB filesystem. Backup release files must be in the Flash filesystem. Valid release files must have a **.rel** extension.
backup	The specified file is a backup release file.

Mode Global Configuration

Usage You can only specify that the release file is on a USB storage device if there is a backup release file already specified in Flash. If you attempt to set the release file on a USB storage device and a backup release file is not specified in Flash, the following error message is displayed:

```
% A backup boot image must be set before setting a current boot
image on USB storage device
```

Examples To run the release file x510-5.4.2A.rel stored on the switch's Flash filesystem the next time the device boots up, use the commands:

```
        awplus# configure terminal
awplus(config)# boot system flash:/x510-5.4.2A.rel
```

To run the release file x510-5.4.2A.rel stored on the switch's USB storage device filesystem the next time the device boots up, use the commands:

```
        awplus# configure terminal
awplus(config)# boot system usb:/x510-5.4.2A.rel
```

To specify the file x510-5.4.2A.rel as the backup to the main release file, use the commands:

```
        awplus# configure terminal
awplus(config)# boot system backup flash:/x510-5.4.2A.rel
```

Related Commands boot config-file
show boot

cd

This command changes the current working directory.

Syntax cd <directory-url>

Parameter	Description
<directory-url>	URL of the directory.

Mode Privileged Exec

Example To change to the directory called images, use the command:

```
awplus# cd images
```

Related Commands dir
pwd
show file systems

copy current-software

This command copies the AlliedWare PlusTM OS software that the device has booted from to a destination file. Specify whether the destination is Flash, or USB storage device, when saving the software to the local filesystem.

Syntax copy current-software <destination-url>

Parameter	Description
<destination-url>	The URL where you would like the current running-release saved. This command creates a file if no file exists with the specified filename. If a file already exists, then the CLI prompts you before overwriting the file. See "URL Syntax and Keyword Usage" on page 7.3 for valid URL syntax.

Mode Privileged Exec

Example To copy the current software as installed in the working directory with the file name my-release.rel, use the command:

```
awplus# copy current-software my-release.rel
```

Related Commands boot system
show boot

copy debug

This command copies a specified debug file to a destination file. Specify whether the destination is Flash or USB storage device when saving the software to the local filesystem.

Syntax

copy debug {<destination-url>debug|flash|nvs|scp|tftpusb}
{<source-url>|debug|flash|nvs|scp|tftpusb}

Parameter	Description
<destination-url>	The URL where you would like the debug output saved. See "URL Syntax and Keyword Usage" on page 7.3 for valid URL syntax.
<source-url>	The URL where the debug output originates. See "URL Syntax and Keyword Usage" on page 7.3 for valid URL syntax.

Mode

Privileged Exec

Example

To copy debug output to the USB storage device with a filename `my-debug`, use the following command:

```
awplus# copy debug usb:mydebug
```

Output

Figure 7-2: CLI prompt after entering the copy debug command

```
Enter source file name []:
```

Related Commands

delete debug
move debug

copy (local)

This command copies a file between local filesystems. This allows you to copy a file stored on Flash memory to or from a different memory type attached to your device, such as a USB storage device. By default, the destination filename is the same as the source file.

Syntax copy <local-source> <local-destination> <filename>

Parameter	Description	
<local-source>	Filesystem where the original file is stored.	
	`flash`	Copies the file from Flash memory.
	`usb`	Copies the file from an attached USB storage device.
<local-destination>	Filesystem where the file is copied to.	
	`flash`	Copies the file to Flash memory.
	`usb`	Copies the file to an attached USB storage device.
<filename>	Filename of the file you are copying.	

Mode Privileged Exec

Example To copy the file newconfig.cfg onto your device's Flash from a USB storage device, use the command:

```
awplus# copy USB flash newconfig.cfg
```

Related Commands

copy (URL)
copy zmodem
show file
show file systems

copy running-config

This command copies the running-config to a destination file, or copies a source file into the running-config. Commands entered in the running-config do not survive a device reboot unless they are saved in a configuration file.

Syntax

```
copy <source-url> running-config

copy running-config <destination-url>

copy running-config startup-config
```

Parameter	Description
<source-url>	The URL of a configuration file. This must be a valid configuration file with a **.cfg** filename extension. Specify this when you want the script in the file to become the new running-config. The URL can contain the following protocols or location words. See "URL Syntax and Keyword Usage" on page 7.3 for valid URL syntax.
<destination-url>	The URL where you would like the current running-config saved. This command creates a file if no file exists with the specified filename. If a file already exists, then the CLI prompts you before overwriting the file. See "URL Syntax and Keyword Usage" on page 7.3 for valid URL syntax.
startup-config	Copies the running-config into the file set as the current startup-config file.

Mode Privileged Exec

Examples To copy the running-config into the startup-config, use the command:

```
awplus# copy running-config startup-config
```

To copy the file layer3.cfg into the running-config, use the command:

```
awplus# copy layer3.cfg running-config
```

To use SCP to copy the running-config as current.cfg to the remote server listening on TCP port 2000, use the command:

```
awplus# copy running-config scp://user@server:2000/
        config_files/current.cfg
```

Related Commands copy startup-config
write file
write memory

copy startup-config

This command copies the startup-config script into a destination file, or alternatively copies a configuration script from a source file into the startup-config file. Specify whether the destination is Flash, or a USB storage device, when loading from the local filesystem.

Syntax copy <source-url> startup-config

copy startup-config <destination-url>

Parameter	Description
<source-url>	The URL of a configuration file. This must be a valid configuration file with a **.cfg** filename extension. Specify this to copy the script in the file into the *startup-config* file. Note that this does not make the copied file the new startup file, so any further changes made in the configuration file are not added to the startup-config file unless you reuse this command. See "URL Syntax and Keyword Usage" on page 7.3 for valid URL syntax.
<destination-url>	The destination and filename that you are saving the startup-config as. This command creates a file if no file exists with the specified filename. If a file already exists, then the CLI prompts you before overwriting the file. See "URL Syntax and Keyword Usage" on page 7.3 for valid URL syntax.

Mode Privileged Exec

Examples To copy the file Layer3.cfg to the startup-config, use the command:

```
awplus# copy Layer3.cfg startup-config
```

To copy the startup-config as the file oldconfig.cfg in the current directory, use the command:

```
awplus# copy startup-config oldconfig.cfg
```

Related Commands copy running-config

copy (URL)

This command copies a file. This allows you to:

- copy files from your device to a remote device
- copy files from a remote device to your device
- copy files stored on Flash memory to or from a different memory type, such as a USB storage device
- create two copies of the same file on your device

Syntax copy <source-url> <destination-url>

Parameter	Description
<source-url>	The URL of the source file. See "URL Syntax and Keyword Usage" on page 7.3 for valid URL syntax.
<destination-url>	The URL for the destination file. See "URL Syntax and Keyword Usage" on page 7.3 for valid URL syntax.

Mode Privileged Exec

Examples To use TFTP to copy the file bob.key into the current directory from the remote server at 10.0.0.1, use the command:

```
awplus# copy tftp://10.0.0.1/bob.key bob.key
```

To use SFTP to copy the file new.cfg into the current directory from a remote server at 10.0.1.2, use the command:

```
awplus# copy sftp://10.0.1.2/new.cfg bob.key
```

To use SCP with the username beth to copy the file old.cfg into the directory config_files on a remote server that is listening on TCP port 2000, use the command:

```
awplus# copy scp://beth@serv:2000/config_files/old.cfg old.cfg
```

To copy the file config.cfg into the current directory from a USB storage device, and rename it to configtest.cfg, use the command:

```
awplus# copy usb:/config.cfg configtest.cfg
```

Related Commands copy (local)
copy zmodem
show file systems

copy zmodem

This command allows you to copy files using ZMODEM using Minicom. ZMODEM works over a serial connection and does not need any interfaces configured to do a file transfer.

Syntax

```
copy <source-url> zmodem
```

```
copy zmodem
```

Parameter	Description
<source-url>	The URL of the source file. See "URL Syntax and Keyword Usage" on page 7.3 for valid URL syntax.

Mode Privileged Exec

Example To copy the local file asuka.key using ZMODEM, use the command:

```
awplus# copy asuka.key zmodem
```

Related Commands

copy (local)
copy (URL)
show file systems

create autoboot

Use this command to create an autoboot.txt file on external media. This command will automatically ensure that the keys and values that are expected in this file are correct. After the file is created the create autoboot command will copy the current release and configuration files across to the external media. The external media is then available to restore a release file and/or a configuration file to the device.

Syntax create autoboot [usb]

Mode Privileged Exec

Example To create an autoboot.txt on external media, use the command:

```
awplus# create autoboot usb
```

Related Commands autoboot enable
show autoboot
show boot

delete

This command deletes files or directories.

Syntax delete [force] [recursive] <url>

Parameter	Description
`force`	Ignore nonexistent filenames and never prompt before deletion.
`recursive`	Remove the contents of directories recursively.
`<url>`	URL of the file to delete. See "URL Syntax and Keyword Usage" on page 7.3 for valid URL syntax.

Mode Privileged Exec

Examples To delete the file temp.cfg from the current directory, use the command:

```
awplus# delete temp.cfg
```

To delete the read-only file one.cfg from the current directory, use the command:

```
awplus# delete force one.cfg
```

To delete the directory old_configs, which is not empty, use the command:

```
awplus# delete recursive old_configs
```

To delete the directory new_configs, which is not empty, without prompting if any read-only files are being deleted, use the command:

```
awplus# delete force recursive new_configs
```

Related Commands erase startup-config
rmdir

delete debug

Use this command to delete a specified debug output file.

Syntax delete debug <source-url>

Parameter	Description
<source-url>	The URL where the debug output originates. See "URL Syntax and Keyword Usage" on page 7.3 for valid URL syntax.

Mode Privileged Exec

Example To delete debug output, use the following command:

```
awplus# delete debug
```

Output Figure 7-3: CLI prompt after entering the delete debug command

```
Enter source file name []:
```

Related Commands copy debug
move debug

dir

This command lists the files on a filesystem. If no directory or file is specified then this command lists the files in the current working directory.

Syntax dir [all] [recursive] [<url>|debug|flash|nvs|usb]

Parameter	Description
`all`	List all files.
`recursive`	List the contents of directories recursively.
`<url>`	URL of the directory or file. If no directory or file is specified, then this command lists the files in the current working directory.
`debug`	Debug root directory
`flash`	Flash memory root directory
`nvs`	NVS memory root directory
`usb`	USB storage device root directory

Mode Privileged Exec

Usage In a stacked environment you can use the CLI on a stack master to access filesystems that are located on another stack member. In this case, when you enter the command, specify the stack member's filesystem by using the following syntax:
<hostname>-<member-id>/
(for example, awplus-1/ for a file or directory on stack member 1, awplus-2/ for member 2 etc).

Examples To list the files in the current working directory, use the command:

```
awplus# dir
```

To list the non-hidden files in the root of the Flash filesystem, use the command:

```
awplus# dir flash
```

To list all the files in the root of the Flash filesystem, use the command:

```
awplus# dir all flash:
```

To list recursively the files in the Flash filesystem, use the command:

```
awplus# dir recursive flash:
```

To list the files within the Flash filesystem for stack member 3, use the command:

```
awplus# dir awplus-3/flash:/
```

Note that you must specify the filesystem, on the stack member (flash in this example).

Related Commands

cd
pwd

edit

This command opens a text file in the AlliedWare PlusTM text editor. Once opened you can use the editor to alter to the file.

If a filename is specified and it already exists, then the editor opens it in the text editor.

If no filename is specified, the editor prompts you for one when you exit it.

Before starting the editor make sure your terminal, terminal emulation program, or Telnet client is 100% compatible with a VT100 terminal. The editor uses VT100 control sequences to display text on the terminal.

For more information about using the editor, including control sequences, see "Using the editor" on page 6.6.

Syntax edit [<filename>]

Parameter	Description
<filename>	Name of a file in the local Flash filesystem.

Mode Privileged Exec

Examples To create and edit a new text file, use the command:

```
awplus# edit
```

To edit the existing configuration file myconfig.cfg stored on your device's Flash memory, use the command:

```
awplus# edit myconfig.cfg
```

Related Commands edit URL
show file

edit URL

This command opens a remote text file as read-only in the AlliedWare PlusTM text editor.

Before starting the editor make sure your terminal, terminal emulation program, or Telnet client is 100% compatible with a VT100 terminal. The editor uses VT100 control sequences to display text on the terminal.

Syntax edit <url>

Parameter	Description
<url>	The URL of the remote file. See "URL Syntax and Keyword Usage" on page 7.3 for valid URL syntax.

Mode Privileged Exec

Example To view the file bob.key stored in the security directory of a TFTP server, use the command:

```
awplus# edit tftp://security/bob.key
```

Related Commands edit
show file

erase startup-config

This command deletes the file that is set as the startup-config file, which is the configuration file that the system runs when it boots up.

At the next restart, the device loads the default configuration file, default.cfg. If default.cfg no longer exists, then the device loads with the factory default configuration. This provides a mechanism for you to return the device to the factory default settings.

Syntax erase startup-config

Mode Privileged Exec

Example To delete the file currently set as the startup-config, use the command:

```
awplus# erase startup-config
```

Related Commands boot config-file
copy running-config
copy startup-config
show boot

license

This command enables the licensed software feature set.

Use the no variant of this command to disable the licensed software feature set.

For feature licenses, contact your authorized distributor or reseller. If a license key expires or a proper key is not installed, some software features will not be available.

Note See the AlliedWare Plus^{TM} datasheet for a list of current feature licenses available by product, and the AlliedWare Plus^{TM} How To notes for information on obtaining them.

Syntax

license <name> <key>

no license [<name>|index <index-number>]

Parameter	Description
<name>	A unique user-defined name for the license. To determine names already in use, use the show license command.
<key>	The encrypted license key to enable this set of software features.
<index-number>	The index number of the software feature. To display the index number, use the show license command.

Mode

Privileged Exec

Usage

Default feature license names are issued along with encrypted license keys by email for you to apply using this command to enable features. These default feature license names can be changed, but must be 15 characters or less in length to be accepted with the issued key.

For example, you may want to change the license name 'AT-FL-SBX9-01' to 'x510 Basic license'. The license name and license index is displayed with the show license command.

Examples

To enable the license name1 with the key 12345678ABCDE123456789ABCDE, use the command:

```
awplus# license name1 12345678ABCDE123456789ABCDE
```

To remove the license name1, use the command:

```
awplus# no license name1
```

Validation Command

show license

mkdir

This command makes a new directory.

Syntax mkdir <url>

Parameter	Description
<url>	URL of the directory that you are creating.

Mode Privileged Exec

Usage The keywords flash, nvs, usb, tftp, scp, sftp and http are reserved for tab completion when using the copy, move, delete, cd and dir command. Keywords flash, nvs, usb, tftp, scp, sftp and http cannot be applied as directory or subdirectory names when using a mkdir command.

Example To make a new directory called images in the current directory, use the command:

```
awplus# mkdir images
```

Related Commands cd
dir
pwd

move

This command renames or moves a file.

Syntax move <source-url> <destination-url>

Parameter	**Description**
<source-url>	The URL of the source file. See "URL Syntax and Keyword Usage" on page 7.3 for valid URL syntax.
<destination-url>	The URL of the destination file. See "URL Syntax and Keyword Usage" on page 7.3 for valid URL syntax.

Mode Privileged Exec

Examples To rename the file temp.cfg to startup.cfg, use the command:

```
awplus# move temp.cfg startup.cfg
```

To move the file temp.cfg from the root of the Flash filesystem to the directory myconfigs, use the command:

```
awplus# move temp.cfg myconfigs/temp.cfg
```

Related Commands delete
edit
show file
show file systems

move debug

This command moves a specified debug file to a destination debug file. Specify whether the destination is Flash or Card when saving the software to the local filesystem.

Syntax

```
move debug {<destination-url>|debug|flash|nvsusb}
      {<source-url>|debug|flash|nvsusb}
```

Parameter	Description
<destination-url>	The URL where you would like the debug output moved to. See "URL Syntax and Keyword Usage" on page 7.3 for valid URL syntax.
<source-url>	The URL where the debug output originates. See "URL Syntax and Keyword Usage" on page 7.3 for valid URL syntax.

Mode Privileged Exec

Example To move debug output onto a USB storage device with a filename my-debug, use the following command:

```
awplus# move debug usb:my-debug
```

Output Figure 7-4: CLI prompt after entering the move debug command

```
Enter source file name []:
```

Related Commands copy debug
delete debug

pwd

This command prints the current working directory.

Syntax pwd

Mode Privileged Exec

Example To print the current working directory, use the command:

```
awplus# pwd
```

Related Commands cd

rmdir

This command removes a directory. The directory must be empty for the command to work unless the optional force keyword is used to remove all subdirectories or files in a directory.

Syntax rmdir [force] <url>

Parameter	Description
`force`	Optional keyword that allows you to delete any directories that are not empty and may contain files or subdirectories.
`<url>`	The URL of the directory.

Mode Privileged Exec

Examples To remove the directory images from the top level of the Flash filesystem, use the command:

```
awplus# rmdir flash:/images
```

To force the removal of directory level1 containing subdirectory level2, use the command:

```
awplus# mkdir level1
awplus# mkdir level1/level2
awplus# rmdir force level1
```

Related Commands cd
dir
mkdir
pwd

show autoboot

This command displays the Autoboot configuration and status.

Syntax show autoboot

Mode Privileged Exec

Example To show the Autoboot configuration and status, use the command:

```
awplus# show autoboot
```

Output Figure 7-5: Example output from the show autoboot command

```
awplus#show autoboot
Autoboot configuration
--------------------------------------------------------------------------------
Autoboot status                             : enabled
Usb storage device file autoboot.txt exists : yes

Restore information on USB drive
Autoboot enable in autoboot.txt             : yes
Restore release file                        : x510-5.4.2A.rel (file exists)
Restore configuration file                  : network_1.cfg (file exists)
```

Figure 7-6: Example output from the show autoboot command when an external media source is not present

```
awplus#show autoboot
Autoboot configuration
--------------------------------------------------------------------------------
Autoboot status                  : enabled
External media source            : usb storage device not found.
```

Related Commands autoboot enable
create autoboot
show boot

show boot

This command displays the current boot configuration.

Syntax show boot

Mode Privileged Exec

Example To show the current boot configuration, use the command:

```
awplus# show boot
```

Output Figure 7-7: Example output from the show boot command with the current boot config set on a USB storage device

```
awplus#show boot
Boot configuration
----------------------------------------------------------------
Current software   : x510-5.4.2A.rel
Current boot image : card:/x510-5.4.2A.rel
Backup  boot image : flash:/x510-5.4.2A.rel
Default boot config: flash:/default.cfg
Current boot config: usb:/my.cfg (file exists)
Backup  boot config: flash:/backup.cfg (file not found)
```

Table 7-1: Parameters in the output of the show boot command

Parameter	Description
`Current software`	The current software release that the device is using.
`Current boot image`	The boot image currently configured for use during the next boot cycle.
`Backup boot image`	The boot image to use during the next boot cycle if the device cannot load the main image.
`Default boot config`	The default startup configuration file. The device loads this configuration script if no file is set as the startup-config file.
`Current boot config`	The configuration file currently configured as the startup-config file. The device loads this configuration file during the next boot cycle if this file exists.
`Backup boot config`	The configuration file to use during the next boot cycle if the main configuration file cannot be loaded.
`Autoboot status`	The status of the Autoboot feature; either enabled or disabled.

Related Commands

autoboot enable
boot config-file
boot system
show autoboot

show file

This command displays the contents of a specified file.

Syntax show file {<filename>|<url>}

Parameter	Description
<filename>	Name of a file on the local Flash filesystem.
<url>	URL of a file.

Mode Privileged Exec

Example To display the contents of the file oldconfig.cfg, which is in the current directory, use the command:

```
awplus# show file oldconfig.cfg
```

Related Commands edit
edit URL
show file systems

show file systems

This command lists the filesystems and their utilization information where appropriate.

If this command is entered on the stack master, it will list the filesystems for all the stack members. A stack member heading is displayed to distinguish the different lists shown for each stack member. If it is entered on a specific stack member, as a host-directed command, it will list the filesystems for only that stack member.

Syntax show file systems

Mode Privileged Exec

Examples To display the filesystems for either a standalone device, or a complete stack, use the command:

```
awplus# show file systems
```

To display the filesystems, use the command:

```
awplus# show file systems
```

To list the filesystem for stack member 3, use the command:

```
awplus# remote-command 3 show file systems
```

Output Figure 7-8: Example output from the show file systems command for a two unit stack.

```
awplus#show file systems

STACK member 1:

 Size(B)  Free(B)  Type    Flags  Prefixes   S/D/V   Lcl/Ntwk  Avail
-----------------------------------------------------------------
  30.0M     6.7M   flash     rw   flash:     static  local      Y
      -        -   system    rw   system:    virtual local      -
 499.0k        0   nvs       rw   nvs:       static  local      Y
      -        -   usbstick  rw   usb:       dynamic local      N
      -        -   tftp      rw   tftp:      -       network    -
      -        -   scp       rw   scp:       -       network    -
      -        -   sftp      ro   sftp:      -       network    -
      -        -   http      ro   http:      -       network    -
-----------------------------------------------------------------

STACK member 2:

 Size(B)  Free(B)  Type    Flags  Prefixes   S/D/V   Lcl/Ntwk  Avail
-----------------------------------------------------------------
  30.0M     6.7M   flash     rw   flash:     static  local      Y
      -        -   system    rw   system:    virtual local      -
 499.0k        0   nvs       rw   nvs:       static  local      Y
      -        -   usbstick  rw   usb:       dynamic local      N
      -        -   tftp      rw   tftp:      -       network    -
      -        -   scp       rw   scp:       -       network    -
      -        -   sftp      ro   sftp:      -       network    -
      -        -   http      ro   http:      -       network    -
.
.
.
```

Table 7-2: Parameters in the output of the **show file systems** command

Parameter	Description
`Size (B) Available`	The total memory available to this filesystem. The units are given after the value and are M for Megabytes or k for kilobytes.
`Free (B)`	The total memory free within this filesystem. The units are given after the value and are M for Megabytes or k for kilobytes.
`Type`	The memory type used for this filesystem: flash, system, nvs, usb, tftp, scp, sftp, or http.
`Flags`	The file setting options: rw (read write), ro (read only).
`Prefixes`	The prefixes used when entering commands to access the filesystems: flash, system, nvs, usbstick, tftp, scp, sftp, or http.
`S/V/D`	The memory type: static, virtual, dynamic.
`Lcl / Ntwk`	Whether the memory is located locally or via a network connection.
`Avail`	Whether the memory is accessible: Y (yes), N (no), - (not appropriate)

Related Commands

edit
edit URL
show file

show license

This command displays information about a specific software license, or all enabled software feature licenses on the device.

Syntax show license [<name>|index <index-number>] [brief]

Parameter	Description
<name>	The license name of the software feature to show information about.
<index-number>	The index number of the software feature to display information about.
brief	Displays a brief summary of license information.

Mode User Exec and Privileged Exec

Examples To display a brief summary of information about all enabled licenses, use the command:

```
awplus# show license brief
```

To display full information about all enabled licenses, use the command:

```
awplus# show license
```

To display full information about the licenses with index number 1, use the command:

```
awplus# show license index 1
```

Output Figure 7-9: Example output from the show license index command

```
awplus#show license index 0
OEM Territory: ATKK
Software Feature Licenses
-------------------------------------------------------------
Index                          : 0
License name                   : Base License
Customer name                  : Base License
Quantity of licenses           : 1
Type of license                : Full
License issue date             : 07-Jul-2000
License expiry date            : N/A
Features included              : VRRP
```

Table 7-3: Parameters in the output of the **show license** command

Parameter	Description
`Index`	Index identifying entry.
`License name`	Name of the license key bundle (case-sensitive).
`Customer name`	Customer name.
`Quantity of licenses`	Quantity of licensed installations.
`Type of license`	Full or Temporary.
`License issue date`	Date the license was generated.
`License expiry date`	Expiry date for temporary license.
`Features included`	List of features included in the license.

Figure 7-10: Example output from the show license brief command

```
awplus#show license brief
OEM Territory: ATKK
Software Feature Licenses
--------------------------------------------------------------
Index License name    Quantity     Customer name
      Type                         Period
--------------------------------------------------------------
0     Base License    1            Base License
      Full                         N/A

Current enabled features for displayed licenses:
 VRRP
```

Table 7-4: Parameters in the output of the **show license** command

Parameter	Description
`Index`	Index identifying entry.
`License name`	Name of the license key bundle (case-sensitive).
`Quantity`	Quantity of licensed installations.
`Customer name`	Customer name.
`Type`	Full or Temporary.
`Period`	Expiry date for temporary license.
`Current enabled features for displayed licenses`	List of features included in the license.

Related Commands license

show running-config

This command displays the current configuration of the device. The output includes all non-default configuration; default settings are not displayed.

You can control the output in any one of the following ways:

- To display only lines that contain a particular word, follow the command with | include word
- To start the display at the first line that contains a particular word, follow the command with | begin word
- To save the output to a file, follow the command with > filename

For more information, see "Controlling "show" Command Output" on page 1.41.

Syntax show running-config

Mode Privileged Exec and Global Configuration

Example To display the current dynamic configuration of your device, use the command:

```
awplus# show running-config
```

Output Figure 7-11: Example output from the show running-config command

```
awplus#sho running-config
!
service password-encryption
!
username manager privilege 15 password 8 $1$bJoVec4D$JwOJGPr7YqoExA0GVasdE0
!
service telnet
!
no clock timezone
ip domain-lookup
!
spanning-tree mode rstp
no platform e2efc
!
interface port1.0.1-1.0.24
 switchport
 switchport mode access
!
!
service telnet
!
no clock timezone
!
!
stack virtual-mac
stack virtual-chassis-id 2111
!
!
ip domain-lookup
!
spanning-tree mode rstp
no platform e2efc
!
interface port1.0.1-1.0.24
 switchport
 switchport mode access
!
interface vlan2
 ip address 172.28.8.210/16
!
ip route 0.0.0.0/0 172.28.0.1
!
line con 0
line vty 0 4
!
end
```

Related Commands copy running-config
show running-config access-list

show running-config access-list

Use this command to show the running system status and configuration details for access-list.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax show running-config access-list

Mode Privileged Exec and Global Configuration

Example To display the running system status and configuration details for access-list, use the command:

```
awplus# show running-config access-list
```

Output Figure 7-12: Example output from the show running-config access-list command

```
!
access-list abc remark annai
access-list abc deny any
access-list abd deny any
!
```

Related Commands copy running-config
show running-config

show running-config as-path access-list

Use this command to show the running system status and configuration details for as-path access-list.

For information on output options, see “Controlling “show” Command Output” on page 1.41.

Syntax show running-config as-path access-list

Mode Privileged Exec and Global Configuration

Example To display the running system status and configuration details for as-path access-list, use the command:

```
awplus# show running-config as-path access-list
```

Output Figure 7-13: Example output from the show running-config as-path access-list command

```
!
ip as-path access-list wer permit knsmk
!
```

Related Commands copy running-config
show running-config

show running-config community-list

Use this command to show the running system status and configuration details for community-lists.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax show running-config community-list

Mode Privileged Exec and Global Configuration

Example To display the running system status and configuration details for community-lists use the command:

```
awplus# show running-config community-list
```

Output Figure 7-14: Example output from the show running-config community list command

```
!
ip community-list standard aspd permit internet
ip community-list expanded cspd deny ljj
ip community-list expanded cspd permit dcv
ip community-list expanded wde permit njhd
ip community-list expanded wer deny sde
```

Related Commands copy running-config
show running-config

show running-config dhcp

Use this command to display the running configuration for DHCP server, DHCP snooping, and DHCP relay.

Syntax show running-config dhcp

Mode Privileged Exec and Global Configuration

Example To display to display the running configuration for DHCP server, DHCP snooping, and DHCP relay:

```
awplus# show running-config dhcp
```

Output Figure 7-15: Example output from the show running-config dhcp command

```
!
#show running-config dhcp
no service dhcp-server
!
service dhcp-snooping
!
interface port1.0.1
 ip dhcp snooping trust
!
interface port1.0.21
 ip dhcp snooping max-bindings 25
 access-group dhcpsnooping
!
interface port2.0.21
 ip dhcp snooping max-bindings 25
 access-group dhcpsnooping
!
interface port2.0.44
 access-group dhcpsnooping
!
interface port3.0.1
 ip dhcp snooping trust
!
interface port3.0.21
 ip dhcp snooping max-bindings 25
!
interface port4.0.24
 access-group dhcpsnooping
!
interface po1
 ip dhcp snooping max-bindings 25
 arp security violation log
!
interface sa1
 ip dhcp snooping max-bindings 25
 access-group dhcpsnooping
 arp security violation log
!
interface vlan100
 ip dhcp snooping
 arp security
!
interface vlan200
 ip dhcp snooping
 arp security
!
```

Related Commands copy running-config
show running-config

show running-config full

Use this command to show the complete status and configuration of the running system.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax show running-config full

Mode Privileged Exec and Global Configuration

Example To display the complete status and configuration of the running system, use the command:

```
awplus# show running-config full
```

Output Figure 7-16: Example output from the show running-config full command

```
awplus#show running-config full
!
no service password-encryption
!
interface lo
ip address 127.0.0.1/8
ipv6 address ::1/128
!
interface vlan1
ip address 10.92.0.16/24
ipv6 address fe80::202:b3ff:fea1:2159/64
!
interface vlan2
ip address 20.10.10.54/24
ipv6 address fe80::200:5eff:fe00:101/64
ipv6 address fe80::202:b3ff:fea1:1567/64
ipv6 address fe80::204:76ff:fee6:6c1c/64
ip rip authentication string abcdefghijklmnop
!
end
```

Related Commands copy running-config
show running-config

show running-config interface

This command displays the current configuration of one or more interfaces on the switch.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax show running-config interface [<interface-list>]
[dot1x|ip igmp|ip multicast|ip |lacp|mstp|rip|rstp|stp]

Parameter	Description
`<interface-list>`	The interfaces or ports to display information about. An interface-list can be: ■ an interface (e.g. `vlan2`), a switch port (e.g. `port1.0.12`), a static channel group (e.g. `sa3`) or a dynamic (LACP) channel group (e.g. `po4`) ■ a continuous range of interfaces, ports, static channel groups or dynamic (LACP) channel groups separated by a hyphen; e.g. `vlan2-8`, or `port1.0.1-1.0.24`, or `sa2-4`, or `po1-3` ■ a comma-separated list of the above; e.g. `port1.0.1,port1.0.8-1.0.24`. Do not mix interface types in a list The specified interfaces must exist.
`dot1x`	Displays running configuration for 802.1X port authentication for the specified interfaces.
`lacp`	Displays running configuration for LACP (Link Aggregation Control Protocol) for the specified interfaces.
`ip igmp`	Displays running configuration for IGMP (Internet Group Management Protocol) for the specified interfaces.
`ip multicast`	Displays running configuration for general multicast settings for the specified interfaces.
`mstp`	Displays running configuration for MSTP (Multiple Spanning Tree Protocol) for the specified interfaces.
`rstp`	Displays running configuration for RSTP (Rapid Spanning Tree Protocol) for the specified interfaces.
`stp`	Displays running configuration for STP (Spanning Tree Protocol) for the specified interfaces.

Mode Privileged Exec and Global Configuration

Examples To display the current running configuration of a switch for VLAN 1, use the command:

```
awplus# show running-config interface vlan1
```

To display the current running configuration of a switch for VLANs 1 and 3-5, use the command:

```
awplus# show running-config interface vlan1,vlan3-vlan5
```

Output Figure 7-17: Example output from the show running-config interface command

```
awplus#sh running-config interface
interface port1.0.1-1.0.24
 switchport
 switchport mode access
!
interface vlan1
 ip address 192.168.1.1/24
 ip rip authentication mode md5
 ip rip authentication string mykey
 ip irdp
!
interface vlan2
 ip address 192.168.2.2/24
 ip rip authentication mode md5
 ip rip authentication key-chain cars
!
```

Related Commands

copy running-config
show running-config

show running-config ip route

Use this command to show the running system static IPv4 route configuration.

For information on output options, see “Controlling “show” Command Output” on page 1.41.

Syntax show running-config ip route

Mode Privileged Exec and Global Configuration

Example To display the running system static IPv4 route configuration, use the command:

```
awplus# show running-config ip route
```

Output Figure 7-18: Example output from the show running-config ip route command

```
!
ip route 3.3.3.3/32 vlan3
ip route 3.3.3.3/32 vlan2
!
```

Related Commands copy running-config
show running-config

show running-config ipv6 access-list

Use this command to show the running system status and configuration for ipv6 access-list.

For information on output options, see “Controlling “show” Command Output” on page 1.41.

Syntax show running-config ipv6 access-list

Mode Privileged Exec and Global Configuration

Example To display the running system status and configuration for ipv6 access-list, use the command:

```
awplus# show running-config ipv6 access-list
```

Output Figure 7-19: Example output from the show running-config ipv6 access-list command

```
!
ipv6 access-list abc permit any
!
```

Related Commands copy running-config
show running-config

show running-config ipv6 prefix-list

Use this command to show the running system status and configuration details for ipv6 prefix-list.

For information on output options, see “Controlling “show” Command Output” on page 1.41.

Syntax show running-config ipv6 prefix-list

Mode Privileged Exec and Global Configuration

Example To display show the running system status and configuration details for ipv6 prefix-list, use the command:

```
awplus# show running-config ipv6 prefix-list
```

Output Figure 7-20: Example output from the show running-config ipv6 prefix-list command

```
!
ipv6 prefix-list sde seq 5 permit any
!
```

Related Commands copy running-config
show running-config

show running-config ipv6 route

Use this command to show the running system static IPv6 route configuration.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax show running-config ipv6 route

Mode Privileged Exec and Global Configuration

Example To display the running system static IPv6 route configuration, use the command:

```
awplus# show running-config ipv6 route
```

Output Figure 7-21: Example output from the show running-config ipv6 route command

```
!
ipv6 route 3e11::/64 lo
ipv6 route 3e11::/64 vlan2
ipv6 route fe80::/64 vlan3
!
```

Related Commands copy running-config
show running-config

show running-config key chain

Use this command to show the running system key-chain related configuration.

For information on output options, see “Controlling “show” Command Output” on page 1.41.

Syntax show running-config key chain

Mode Privileged Exec and Global Configuration

Example To display the running system key-chain related configuration, use the command:

```
awplus# show running-config key chain
```

Output Figure 7-22: Example output from the show running-config key chain command

```
!
key chain 12
key 2
key-string 234
!
key chain 123
key 3
key-string 345
!
```

Related Commands copy running-config
show running-config

show running-config lldp

This command shows the current running configuration of LLDP.

Syntax show running-config lldp

Mode Privileged Exec and Global Configuration

Example To display the current configuration of LLDP, use the command:

```
awplus# show running-config lldp
```

Output Figure 7-23: Example output from the show running-config lldp command

```
awplus#show running-config lldp

lldp notification-interval 10
lldp timer 20
!
interface port1.0.1
 lldp notifications
 lldp tlv-select port-description
 lldp tlv-select system-name
 lldp tlv-select system-description
 lldp tlv-select management-address
 lldp transmit receive
```

Related Commands show lldp
show lldp interface

show running-config prefix-list

Use this command to show the running system status and configuration details for prefix-list.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax show running-config prefix-list

Mode Privileged Exec and Global Configuration

Example To display the running system status and configuration details for prefix-list, use the command:

```
awplus# show running-config prefix-list
```

Output Figure 7-24: Example output from the show running-config prefix-list command

```
!
ip prefix-list abc seq 5 permit any
ip prefix-list as description annai
ip prefix-list wer seq 45 permit any
!
```

Related Commands copy running-config
show running-config

show running-config

show running-config router-id

Use this command to show the running system global router ID configuration.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax show running-config router-id

Mode Privileged Exec and Global Configuration

Example To display the running system global router ID configuration, use the command:

```
awplus# show running-config router-id
```

Output Figure 7-25: Example output from the show running-config router-id command

```
!
router-id 3.3.3.3
!
```

Related Commands copy running-config
show running-config

show running-config security-password

This command displays the configuration settings for the various security-password rules. If a default parameter is used for a security-password rule, therefore disabling that rule, no output is displayed for that feature.

For information on output options, see “Controlling “show” Command Output” on page 1.41.

Syntax show running-config security-password

Mode Privileged Exec and Global Configuration

Example To display the current security-password rule settings in the running-config, use the command:

```
awplus# show running-config security-password
```

Output Figure 7-26: Example output from the show running-config security-password command

```
security-password minimum-length 8
security-password minimum-categories 3
security-password history 4
security-password lifetime 30
security-password warning 3
security-password forced-change
```

Related Commands show security-password configuration
show security-password user

show running-config switch

Use this command to show the running system status and configuration details for a given switch.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax show running-config switch <switch-protocol>

Parameter	Description
<switch-protocol>	dot1x | mstp | rstp | stp
dot1x	802.1X Port-Based Authentication
mstp	Multiple Spanning Tree Protocol (MSTP)
rstp	Rapid Spanning Tree Protocol (RSTP)
stp	Spanning Tree Protocol (RSTP)

Mode Privileged Exec and Global Configuration

Example To display the running system status and configuration details for a given switch, use the command:

```
awplus# show running-config switch stp
```

Output Figure 7-27: Example output from the show running-config switch command

```
!
bridge 6 ageing-time 45
bridge 6 priority 4096
bridge 6 max-age 7
```

Related Commands copy running-config
show running-config

show running-config switch lacp

Use this command to show the running system LACP related configuration.

For information on output options, see “Controlling “show” Command Output” on page 1.41.

Syntax show running-config switch lacp

Mode Privileged Exec and Global Configuration

Example To display the running system LACP related configuration, use the command:

```
awplus# show running-config switch lacp
```

Output Figure 7-28: Example output from the show running-config switch lacp command

```
!
lacp system-priority 23
!
```

Related Commands copy running-config
show running-config

show running-config switch radius-server

Use this command to show the running system RADIUS-server related configuration.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax show running-config switch radius-server

Mode Privileged Exec and Global Configuration

Example To display the running system radius-server related configuration, use the command:

```
awplus# show running-config switch radius-server
```

Output Figure 7-29: Example output from the show running-config switch radius-server command

```
!
radius-server key abc
!
```

Related Commands copy running-config
show running-config

show running-config switch vlan

Use this command to show the running system VLAN related configuration.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax show running-config switch vlan

Mode Privileged Exec and Global Configuration

Example To display the running system VLAN related configuration, use the command:

```
awplus# show running-config switch vlan
```

Output Figure 7-30: Example output from the show running-config switch vlan command

```
!
vlan database
vlan 4 bridge 2 name VLAN0004
vlan 4 bridge 2 state enable
```

Related Commands copy running-config
show running-config

show startup-config

This command displays the contents of the start-up configuration file, which is the file that the device runs on start-up.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax show startup-config

Mode Privileged Exec

Example To display the contents of the current start-up configuration file, use the command:

```
awplus# show startup-config
```

Output Figure 7-31: Example output from the show startup-config command

```
awplus#show startup-config
!
service password-encryption
!
username manager privilege 15 password 8 $1$bJoVec4D$JwOJGPr7YqoExA0GVasdE0
!
no service ssh
!
service telnet
!
service http
!
no clock timezone
.
.
.
line con 0
line vty 0 4
!
end
```

Related Commands boot config-file
copy running-config
copy startup-config
erase startup-config
show boot

show version

This command displays the version number and copyright details of the current AlliedWare PlusTM OS your device is running.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax show version

Mode User Exec and Privileged Exec

Example To display the version details of your currently installed software, use the command:

```
awplus# show version
```

Output Figure 7-32: Example output from the show version command

```
awplus#show version

AlliedWare Plus (TM) 5.4.2 12/12/11 12:13:19

Build name : x510-5.4.2A.rel
Build date : Wed Dec 2 12:13:19 NZDT 2011
Build type : RELEASE
 NET-SNMP SNMP agent software
 (c) 1996, 1998-2000 The Regents of the University of California.
     All rights reserved;
 (c) 2001-2003, Networks Associates Technology, Inc. All rights reserved;
 (c) 2001-2003, Cambridge Broadband Ltd. All rights reserved;
 (c) 2003, Sun Microsystems, Inc. All rights reserved.
 RSA Data Security, Inc. MD5 Message-Digest Algorithm
 (c) 1991-2, RSA Data Security, Inc. Created 1991. All rights reserved.
 Libedit Library
 (c) 1992, 1993 The Regents of the University of California.
     All rights reserved.
 OpenSSL Library
 Copyright (c) 1998-2008 The OpenSSL Project. All rights reserved.
 Original SSLeay License
 Copyright (C) 1995-1998 Eric Young (eay@cryptsoft.com).
 sFlow(R) Agent Software
 Copyright (c) 2002-2006 InMon Corp.
 DHCP Library
 Copyright (c) 2004-2010 by Internet Systems Consortium, Inc;
 Copyright (c) 1995-2003 by Internet Software Consortium.
 Application Interface Specification Framework
 Copyright (c) 2002-2004 MontaVista Software, Inc;
 Copyright (c) 2005-2010 Red Hat, Inc.
 Hardware Platform Interface Library
 Copyright (c) 2003, Intel Corporation;
 Copyright (C) IBM Corp. 2003-2007.
 Corosync Cluster Engine
 Copyright (c) 2002-2004 MontaVista Software, Inc. All rights reserved.
 File Utility Library
 Copyright (c) Ian F. Darwin 1986-1987, 1989-1992, 1994-1995.
 Software written by Ian F. Darwin and others;
 maintained 1994- Christos Zoulas.

Portions of this product are covered by the GNU GPL, source code may be
downloaded from: http://www.alliedtelesis.co.nz/support/gpl/awp.html
```

Related Commands boot system
show boot

write file

This command copies the running-config into the file that is set as the current startup-config file. This command is a synonym of the write memory and copy running-config startup-config commands.

Syntax write [file]

Mode Privileged Exec

Example To write configuration data to the start-up configuration file, use the command:

```
awplus# write file
```

Related Commands copy running-config
write memory
show running-config

write memory

This command copies the running-config into the file that is set as the current startup-config file. This command is a synonym of the write file and copy running-config startup-config commands.

Syntax write [memory]

Mode Privileged Exec

Example To write configuration data to the start-up configuration file, use the command:

```
awplus# write memory
```

Related Commands copy running-config
write file
show running-config

write terminal

This command displays the current configuration of the device. This command is a synonym of the show running-config command.

Syntax write terminal

Mode Privileged Exec

Example To display the current configuration of your device, use the command:

```
awplus# write terminal
```

Related Commands show running-config

Chapter 8: System Configuration and Monitoring Commands

Command List

This chapter provides an alphabetical reference of commands for configuring and monitoring the system.

banner exec

This command configures the User Exec mode banner that is displayed on the console after you login. The **banner exec default** command restores the User Exec banner to the default banner. Use the **no banner exec** command to disable the User Exec banner and remove the default User Exec banner.

Syntax

```
banner exec <banner-text>

banner exec default

no banner exec
```

Default By default, the AlliedWare Plus^{TM} version and build date is displayed at console login, such as:

```
AlliedWare Plus (TM) 5.4.1 07/27/10 00:44:25
```

Mode Global Configuration

Examples To configure a User Exec mode banner after login, enter the following commands:

```
awplus#configure terminal
awplus(config)#banner exec enable to move to Priv Exec mode
awplus(config)#exit
awplus#exit

awplus login: manager
Password:
enable to move to Priv Exec mode
awplus>
```

To restore the default User Exec mode banner after login, enter the following commands:

```
awplus#configure terminal
awplus(config)#banner exec default
awplus(config)#exit
awplus#exit

awplus login: manager
Password:

AlliedWare Plus (TM) 5.4.1 11/14/10 13:03:59

awplus>
```

To remove the User Exec mode banner after login, enter the following commands:

```
awplus#configure terminal
awplus(config)#banner exec default
awplus(config)#exit
awplus#exit

awplus login: manager
Password:
awplus>
```

Related Commands banner login (system)
banner motd

banner login (system)

This command configures the login banner that is displayed on the console when you login. The login banner is displayed on all connected terminals. The login banner is displayed after the MOTD (Message-of-the-Day) banner and before the login username and password prompts.

Use the **no banner login** command to disable the login banner.

Syntax

```
banner login

no banner login
```

Default By default, no login banner is displayed at console login.

Mode Global Configuration

Examples To configure a login banner to be displayed when you login, enter the following commands:

```
awplus#configure terminal
awplus(config)#banner login
Type CNTL/D to finish.
authorised users only
awplus(config)#exit
awplus#exit

authorised users only
awplus login: manager
Password:

AlliedWare Plus (TM) 5.4.1 11/14/10 13:03:59

awplus>
```

To remove the login banner, enter the following commands:

```
awplus#configure terminal
awplus(config)#no banner login
awplus(config)#exit
awplus#exit

awplus login: manager
Password:

awplus>
```

Related Commands banner exec
banner motd

banner motd

Use this command to change the text MOTD (Message-of-the-Day) banner displayed before login. The MOTD banner is displayed on all connected terminals. The MOTD banner is useful for sending messages that affect all network users, for example, any imminent system shutdowns.

Use the **no** variant of this command to not display a text MOTD (Message-of-the-Day) banner on login.

Syntax

```
banner motd <motd-text>
```

```
no banner motd
```

Default By default, the switch displays the AlliedWare Plus^{TM} OS version and build date before login.

Mode Global Configuration

Examples To configure a MOTD banner to be displayed when you login, enter the following commands:

```
awplus>enable
awplus#configure terminal
awplus(config)#banner motd system shutdown at 6pm
awplus(config)#exit
awplus#exit

system shutdown at 6pm
awplus login: manager
Password:

AlliedWare Plus (TM) 5.4.1 11/14/10 13:03:59
```

To remove the login banner, enter the following commands:

```
awplus>enable
awplus#configure terminal
awplus(config)#no banner motd
awplus(config)#exit
awplus#exit

awplus login: manager
Password:

AlliedWare Plus (TM) 5.4.1 11/14/10 13:03:59

awplus>
```

Related Commands banner exec
banner login (system)

clock set

This command sets the time and date for the system clock.

Syntax

```
clock set <hh:mm:ss> <day> <month> <year>
```

Parameter	Description
<hh:mm:ss>	Local time in 24-hour format
<day>	Day of the current month `<1-31>`
<month>	The first three letters of the current month.
<year>	Current year `<2000-2035>`

Mode Privileged Exec

Usage Configure the timezone before setting the local time. Otherwise, when you change the timezone, the device applies the new offset to the local time.

Note If Network Time Protocol (NTP) is enabled, then you cannot change the time or date using this command. NTP maintains the clock automatically using an external time source. If you wish to manually alter the time or date, you must first disable NTP.

Example To set the time and date on your system to 2pm on the 2nd of April 2007, use the command:

```
awplus# clock set 14:00:00 2 apr 2007
```

Related Commands clock timezone

clock summer-time date

This command defines the start and end of summertime for a specific year only, and specifies summertime's offset value to Standard Time for that year.

The **no** variant of this command removes the device's summertime setting. This clears both specific summertime dates and recurring dates (set with the clock summer-time recurring command on page 8.9).

By default, the device has no summertime definitions set.

Syntax

```
clock summer-time <timezone-name> date <start-day> <start-month>
   <start-year> <start-time> <end-day> <end-month> <end-year>
   <end-time> <1-180>
```

```
no clock summer-time
```

Parameter	Description
<timezone-name>	A description of the summertime zone, up to 6 characters long.
date	Specifies that this is a date-based summertime setting for just the specified year.
<start-day>	Day that the summertime starts, in the range 1-31.
<start-month>	First three letters of the name of the month that the summertime starts.
<start-year>	Year that summertime starts, in the range 2000-2035.
<start-time>	Time of the day that summertime starts, in the 24-hour time format HH:MM.
<end-day>	Day that summertime ends, in the range 1-31.
<end-month>	First three letters of the name of the month that the summertime ends.
<end-year>	Year that summertime ends, in the range 2000-2035.
<end-time>	Time of the day that summertime ends, in the 24-hour time format HH:MM.
<1-180>	The offset in minutes.

Mode Global Configuration

Examples To set a summertime definition for New Zealand using NZST (UTC+12:00) as the standard time, and NZDT (UTC+13:00) as summertime, with the summertime set to begin on the 1st October 2007 and end on the 18th of March 2008:

```
awplus(config)# clock summer-time NZDT date 1 oct 2:00 2007 18
                mar 2:00 2008 60
```

To remove any summertime settings on the system, use the command:

```
awplus(config)# no clock summer-time
```

Related Commands clock summer-time recurring
clock timezone

clock summer-time recurring

This command defines the start and end of summertime for every year, and specifies summertime's offset value to Standard Time.

The **no** variant of this command removes the device's summertime setting. This clears both specific summertime dates (set with the clock summer-time date command on page 8.7) and recurring dates.

By default, the device has no summertime definitions set.

Syntax

```
clock summer-time <timezone-name> recurring <start-week> <start-day>
    <start-month> <start-time> <end-week> <end-day> <end-month>
    <end-time> <1-180>
```

```
no clock summer-time
```

Parameter	Description
<timezone-name>	A description of the summertime zone, up to 6 characters long.
`recurring`	Specifies that this summertime setting applies every year from now on.
<start-week>	Week of the month when summertime starts, in the range 1-5. The value 5 indicates the last week that has the specified day in it for the specified month. For example, to start summertime on the last Sunday of the month, enter **5** for *<start-week>* and **sun** for *<start-day>*.
<start-day>	Day of the week when summertime starts. Valid values are **mon**, **tue**, **wed**, **thu**, **fri**, **sat** or **sun**.
<start-month>	First three letters of the name of the month that summertime starts.
<start-time>	Time of the day that summertime starts, in the 24-hour time format HH:MM.
<end-week>	Week of the month when summertime ends, in the range 1-5. The value 5 indicates the last week that has the specified day in it for the specified month. For example, to end summertime on the last Sunday of the month, enter **5** for *<end-week>* and **sun** for *<end-day>*.
<end-day>	Day of the week when summertime ends. Valid values are **mon**, **tue**, **wed**, **thu**, **fri**, **sat** or **sun**.
<end-month>	First three letters of the name of the month that summertime ends.
<end-time>	Time of the day that summertime ends, in the 24-hour time format HH:MM.
<1-180>	The offset in minutes.

Mode Global Configuration

Examples To set a summertime definition for New Zealand using NZST (UTC+12:00) as the standard time, and NZDT (UTC+13:00) as summertime, with summertime set to start on the 1st Sunday in October, and end on the 3rd Sunday in March, use the command:

```
awplus(config)# clock summer-time NZDT recurring 1 sun oct 2:00
                3 sun mar 2:00 60
```

To remove any summertime settings on the system, use the command:

```
awplus(config)# no clock summer-time
```

Related Commands clock summer-time date
clock timezone

clock timezone

This command defines the device's clock timezone. The timezone is set as a offset to the UTC.

The **no** variant of this command resets the system time to UTC.

By default, the system time is set to UTC.

Syntax

```
clock timezone <timezone-name> {minus|plus} [<0-13>|<0-12>:<00-59>]

no clock timezone
```

Parameter	Description
<timezone-name>	A description of the timezone, up to 6 characters long.
minus or plus	The direction of offset from UTC. The **minus** option indicates that the timezone is behind UTC. The **plus** option indicates that the timezone is ahead of UTC.
<0-13>	The offset in hours or from UTC.
<0-12>:<00-59>	The offset in hours or from UTC.

Mode Global Configuration

Usage Configure the timezone before setting the local time. Otherwise, when you change the timezone, the device applies the new offset to the local time.

Examples To set the timezone to New Zealand Standard Time with an offset from UTC of +12 hours, use the command:

```
awplus(config)# clock timezone NZST plus 12
```

To set the timezone to Indian Standard Time with an offset from UTC of +5:30 hours, use the command:

```
awplus(config)# clock timezone NZST plus 5:30
```

To set the timezone back to UTC with no offsets, use the command:

```
awplus(config)# no clock timezone
```

Related Commands clock set
clock summer-time date
clock summer-time recurring

continuous-reboot-prevention

Use this command to enable and to configure the continuous reboot prevention feature. Continuous reboot prevention allows the user to configure the time period during which reboot events are counted, the maximum number of times the switch can reboot within the specified time period, referred to as the threshold, and the action to take if the threshold is exceeded.

Use the **no** variant of this command to disable the continuous reboot prevention feature or to return the **period**, **threshold** and **action** parameters to the defaults.

Syntax

```
continuous-reboot-prevention enable

continuous-reboot-prevention [period <0-604800>] [threshold <1-10>]
    [action [linkdown|logonly|stopreboot]]

no continuous-reboot-prevention enable

no continuous-reboot-prevention [period] [threshold] [action]}
```

Parameter	Description	
`enable`	Enable the continuous reboot prevention feature.	
`period`	Set the period of time in which reboot events are counted.	
	<0-604800>	Period value in seconds. The default is 600.
`threshold`	Set the maximum number of reboot events allowed in the specified period.	
	<1-10>	Threshold value. The default is 1.
`action`	Set the action taken if the threshold is exceeded.	
	`linkdown`	Reboot procedure continues and all switch ports and stack ports stay link-down. The reboot event is logged. This is the default action.
	`logonly`	Reboot procedure continues normally and the reboot event is logged.
	`stopreboot`	Reboot procedure stops until the user enters the key "c" via the CLI. Normal reboot procedure then continues and the reboot event is logged.

Default Continuous reboot prevention is disabled by default. The default `period` value is `600`, the default `threshold` value is `1` and the default `action` is `linkdown`.

Mode Global Configuration

Examples To enable continuous reboot prevention, use the commands:

```
        awplus# configure terminal
awplus(config)# continuous-reboot-prevention enable
```

To set the period to 500 and action to stopreboot, use the commands:

```
         awplus#  configure terminal

 awplus(config)#  continuous-reboot-prevention period 500
                  action stopreboot
```

To return the period and action to the defaults and keep the continuous reboot prevention feature enabled, use the commands:

```
         awplus#  configure terminal

 awplus(config)#  no continuous-reboot-prevention period action
```

To disable continuous reboot prevention, use the commands:

```
         awplus#  configure terminal

 awplus(config)#  no continuous-reboot-prevention enable
```

Related Commands

show continuous-reboot-prevention
show reboot history
show tech-support

ecofriendly led

Use this command to enable the eco-friendly feature which turns off power to the port LEDs.

Use the **no** variant of this command to disable the eco-friendly feature.

Syntax

```
ecofriendly led
```

```
no ecofriendly led
```

Default The eco-friendly feature is disabled by default.

Mode Global Configuration

Usage When the eco-friendly feature is enabled, a change in port status will not affect the display of the associated LED. When the eco-friendly feature is disabled and power is returned to port LEDs, the LEDs will correctly show the current state of the ports.

In a stack environment, enabling the eco-friendly feature on the stack master will apply the feature to every member of the stack.

For an example of how to configure a trigger to enable the eco-friendly feature, see "Turn Off Power to Port LEDs" on page 71.7.

Example To enable the eco-friendly feature which turns off power to all port LEDs, use the following commands:

```
        awplus# configure terminal
awplus(config)# ecofriendly led
```

To disable the eco-friendly feature, use the following command:

```
        awplus# configure terminal
awplus(config)# no ecofriendly led
```

Related Commands show ecofriendly

hostname

This command sets the name applied to the device as shown at the prompt. The hostname is:

- displayed in the output of the show system command
- displayed in the CLI prompt so you know which device you are configuring
- stored in the MIB object sysName

Use the **no** variant of this command to reset the hostname to the default (`awplus`).

On a stack, after the stack master is elected, the master will have a host name: `awplus` by default, and this also becomes the name of the stack. Individual stack members (excluding the master) will have a host name that is the stack name hyphenated with a numeric suffix. For example, `awplus-1`, `awplus-2` and so on.

The hostname command can then be used to change the stack name and the stack master's host name. For example, for the hostname `Lab` the stack master's host name will be `Lab` and the other stack members will have host names `Lab-1`, `Lab-2` and so on.

In case of stack master fail-over, or stack split, the new stack will use the previous stack name as its host name and the stack name, unless it is changed by executing hostname command on the new stack master.

Use the **no** variant of this command to revert the hostname setting to its default (`awplus`).

Syntax

```
hostname <hostname>

no hostname [<hostname>]
```

Parameter	Description
<hostname>	Specifies the network name of the system.

Default The default hostname is `awplus`.

Mode Global Configuration

Usage To specify or modify the host name, use the hostname global configuration command. The host name is used in prompts and default configuration filenames.

The name must also follow the rules for ARPANET host names. The name must start with a letter, end with a letter or digit, and use only letters, digits, and hyphens. Refer to RFC 1035.

Example To set the system name to `HQ-Sales`, use the command:

```
        awplus# configure terminal
awplus(config)# hostname HQ-Sales
```

This changes the prompt to:

```
HQ-Sales(config)#
```

To revert to the default hostname `awplus`, use the command:

```
        awplus# configure terminal
awplus(config)# no hostname
```

This changes the prompt to:

```
awplus(config)#
```

Related Commands show system

max-fib-routes

This command now enables you to control the maximum number of FIB routes configured. It operates by providing parameters that enable you to configure preset maximums and warning message thresholds. The operation of these parameters is explained in the Parameter / Descriptions table shown below.

Note To set static routes, use the max-static-routes command on page 8.17.

Use the **no** variant of this command to set the maximum number of fib routes to the default of 4294967294 fib routes.

Syntax

```
max-fib-routes <1-4294967294>

no max-fib-routes
```

Syntax

```
max-fib-routes <1-4294967294> [<1-100>|warning-only]

no max-fib-routes
```

Parameter	Description
`max-fib-routes`	This is a the maximum number of routes that can be stored in the switch's Forwarding Information dataBase. In practice, other practical system limits would prevent this maximum being reached.
<1-4294967294>	The allowable configurable range for setting maximum the number of FIB-routes.
<1-100>	This parameter enables you to optionally apply a percentage value. This percentage will be based on the maximum number of FIB routes you have specified. This will cause a warning message to appear when your routes reach your specified percentage value. Routes can continue to be added until your configured maximum value is reached.
`warning-only`	This parameter enables you to optionally apply a warning message. If you set this option a warning message will appear if your maximum configured value configured. Routes can continue to be added until your switch reaches either the maximum capacity value of 4294967294, or a practical system limit.

Default The default number of fib routes is the maximum number of fib routes (4294967294).

Mode Global Configuration

Examples To set the maximum number of dynamic routes to 2000 and warning threshold of 75%, use the following commands:

```
        awplus# config terminal
awplus(config)# max-fib-routes 2000 75
```

max-static-routes

Use this command to set the maximum number of static routes, excluding FIB (Forwarding Information Base) routes. Note that FIB routes are set and reset using max-fib-routes.

Use the **no** variant of this command to set the maximum number of static routes to the default of 1000 static route.

Note To set dynamic FIB routes, use the max-fib-routes command on page 8.16.

Syntax

```
max-static-routes <1-1000>
```

```
no max-static-routes
```

Default The default number of static routes is the maximum number of static routes (1000).

Mode Global Configuration

Example To reset the maximum number of static routes to the default maximum, use the command:

```
        awplus# configure terminal
awplus(config)# no max-static-routes
```

Related Commands max-fib-routes

no debug all

This command disables the debugging facility for all features on your device. This stops the device from generating any diagnostic debugging messages.

The debugging facility is disabled by default.

Syntax

```
no debug all [dot1x|ipv6|nsm|vrrp]
```

Parameter	Description
`dot1x`	Turns off all debugging for IEEE 802.1X port-based network access-control.
`ipv6`	Turns off all debugging for IPv6 (Internet Protocol version 6).
`nsm`	Turns off all debugging for the NSM (Network Services Module).
`vrrp`	Turns off all debugging for VRRP (Virtual Router Redundancy Protocol).

Mode Global Configuration and Privileged Exec

Example To disable debugging for all features, use the command:

```
awplus# no debug all
```

To disable all 802.1X debugging, use the command:

```
awplus# no debug all
```

To disable all IPv6 debugging, use the command:

```
awplus# no debug all
```

To disable all NSM debugging, use the command:

```
awplus# no debug all
```

To disable all VRRP debugging, use the command:

```
awplus# no debug all vrrp
```

Related Commands undebug all

reboot

This command halts and performs a cold restart (also known as reboot or reload) on either a stand-alone device, a stack, or a specified stack member. It displays a confirmation request before restarting any devices.

Syntax

```
reboot [stack-member <1-4>]

reload [stack-member <1-4>]
```

Parameter	Description
`stack-member`	Restart the specified stack member.
`<1-8>`	The ID of the stack member to be restarted.
`<1-4>`	The ID of the stack member to be restarted.

Mode

Privileged Exec

Usage

The **reboot** and **reload** commands perform the same action.

When restarting the whole stack, you can either use this reboot command to reboot all stack members immediately, or to minimize downtime, reboot the stack members in a rolling sequence by using the reboot rolling command on page 78.7.

Examples

To restart the stand-alone device, use the command:

```
awplus# reboot
reboot system? (y/n): y
```

To restart all devices in the stack, use the command:

```
awplus# reboot
Are you sure you want to reboot the whole stack? (y/n): y
```

To restart stack member 3, use the command:

```
awplus# reboot stack-member 3
reboot stack-member 3 system? (y/n): y
```

If the specified stack member ID does not exist in the current stack, the command is rejected.

Related Commands

reboot rolling
reload rolling

reload

This command performs the same function as the reboot command on page 8.19.

show clock

This command displays the system's current configured local time and date. It also displays other clock related information such as timezone and summertime configuration.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax

```
show clock
```

Mode

User Exec and Privileged Exec

Example

To display the system's current local time, use the command:

```
awplus# show clock
```

Output

Figure 8-1: Example output from the **show clock** command for a switch using New Zealand time

```
Local Time: Mon,  6 Aug 2007 13:56:06 +1200
UTC Time:   Mon,  6 Aug 2007 01:56:06 +0000
Timezone: NZST
Timezone Offset: +12:00
Summer time zone: NZDT
Summer time starts: Last Sunday in September at 02:00:00
Summer time ends: First Sunday in April at 02:00:00
Summer time offset: 60 mins
Summer time recurring: Yes
```

Table 8-1: Parameters in the output of the **show clock** command

Parameter	Description
`Local Time`	Current local time.
`UTC Time`	Current UTC time.
`Timezone`	The current configured timezone name.
`Timezone Offset`	Number of hours offset to UTC.
`Summer time zone`	The current configured summertime zone name.
`Summer time starts`	Date and time set as the start of summer time.
`Summer time ends`	Date and time set as the end of summer time.
`Summer time offset`	Number of minutes that summer time is offset from the system's timezone.
`Summer time recurring`	Whether the device will apply the summer time settings every year or only once.

Related Commands

clock set
clock summer-time date
clock summer-time recurring
clock timezone

show continuous-reboot-prevention

This command displays the current continuous reboot prevention configuration.

Syntax

```
show continuous-reboot-prevention
```

Mode

User Exec and Privileged Exec

Examples

To show the current continuous reboot prevention configuration, use the command:

```
awplus# show continuous-reboot-prevention
```

Output

Figure 8-2: Example output from the **show continuous-reboot-prevention** command

```
---------------------------------------------
Continuous reboot prevention
---------------------------------------------
status=disabled
period=600
threshold=1
action=linkdown
---------------------------------------------
```

Related Commands

continuous-reboot-prevention
show reboot history

show cpu

This command displays a list of running processes with their CPU utilization.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax

```
show cpu [<1-8>] [sort {thrds|pri|sleep|runtime}]
```

Parameter	Description	
`<1-8>`	The stack member to display output for.	
`sort`	Whether to sort the list by a specified field. If you do not specify this, then the list is sorted by percentage CPU utilization.	
	`thrds`	The list is sorted by the number of threads.
	`pri`	The list is sorted by the process priority.
	`sleep`	The list is sorted by the average time sleeping.
	`runtime`	The list is sorted by the runtime of the process.

Mode User Exec and Privileged Exec

Usage Entering this command on the stack master will display the information of all the stack members. A stack member heading will distinguish the different information for every stack member device.

Entering this command for a specific stack member (as a host-directed command) will display the information for that stack member.

Examples To show the CPU utilization of current processes and sorting them by the number of threads the processes are using, use the command:

```
awplus# show cpu sort thrds
```

Note that in a stack environment, executing this command on the stack master will show CPU utilization for all stack members.

To show CPU utilization for a specific stack member (in this case stack member 3), use the following command:

```
awplus# show cpu 3
```

Figure 8-3: Example output from the **show cpu** command

```
Stack member 1:

CPU averages:
 1 second: 5%, 20 seconds: 0%, 60 seconds: 0%
System load averages:
 1 minute: 0.00, 5 minutes: 0.00, 15 minutes: 0.00
Current CPU load:
 userspace: 4%, kernel: 1%, interrupts: 0% iowaits: 0%

user processes
==============
  pid name            thrds  cpu%   pri state sleep% runtime
 1532 hostd               1   1.9    20   run     0   103
 1113 exfx               18   0.9    20 sleep     0  3374
 1225 aisexec            44   0.9    -2 sleep     0  2290
 1630 mstpd               1   0.9    20 sleep     0    86
    1 init                1   0.0    20 sleep     0   799
 6149 sh                  1   0.0    20 sleep     0     0
 6150 corerotate          1   0.0    20 sleep     0     0
  801 syslog-ng           1   0.0    20 sleep     0   287
  807 klogd               1   0.0    20 sleep     0     1
  858 inetd               1   0.0    20 sleep     0    21
  868 portmap             1   0.0    20 sleep     0     0
  879 crond               1   0.0    20 sleep     0     1
 1038 openhpid           10   0.0    20 sleep     0   161
 1057 hpilogd             1   0.0    20 sleep     0     0
 1147 stackd              1   0.0    20 sleep     0    10
 1170 hsl                 1   0.0    20 sleep     0     1
 1258 rpc.statd           1   0.0    20 sleep     0     0
 1262 rpc.statd           1   0.0    20 sleep     0     0
 1268 rpc.mountd          1   0.0    20 sleep     0     0
 1361 automount           1   0.0    20 sleep     0    84
 1395 ntpd                1   0.0    20 sleep     0    18
 1440 authd               1   0.0    20 sleep     0    89
 1463 bgpd                1   0.0    20 sleep     0    88
 1483 cntrd               1   0.0    20 sleep     0    89
 1509 epsrd               1   0.0    20 sleep     0    90
 1560 imi                 1   0.0    20 sleep     0    87
 1581 irdpd               1   0.0    20 sleep     0    90
 1603 lacpd               1   0.0    20 sleep     0    86
 1653 nsm                 1   0.0    20 sleep     0   111
  1700 pdmd                1   0.0    20 sleep     0    88
 1722 pimd                1   0.0    20 sleep     0    87
 1743 ripd                1   0.0    20 sleep     0    90
 1765 ripngd              1   0.0    20 sleep     0    88
 1786 rmond               1   0.0    20 sleep     0    91
 1798 sshd                1   0.0    20 sleep     0     0
 1905 atlgetty            1   0.0    20 sleep     0     0
 1906 getty               1   0.0    20 sleep     0     0

kernel threads
==============
  pid name             cpu%   pri state sleep% runtime
   87 aio/0             0.0    15 sleep     0     0
    5 events/0          0.0    15 sleep     0   575
  673 fsl-cpm-spi.1     0.0    15 sleep     0     0
   58 kblockd/0         0.0    15 sleep     0     0
    6 khelper           0.0    15 sleep     0     1
  667 kmmcd             0.0    15 sleep     0     0
    3 ksoftirqd/0       0.0    15 sleep     0    78
   86 kswapd0           0.0    15 sleep     0     0
    2 kthreadd          0.0    15 sleep     0     0
  989 loop0             0.0     0 sleep     0    16
  648 mtdblockd         0.0    15 sleep     0     5
   84 pdflush           0.0    20 sleep     0     0
  732 rpciod/0          0.0    15 sleep     0     0
  689 w1_control        0.0    15 sleep     0     2
    4 watchdog/0        0.0  -100 sleep     0     0
  768 jffs2_gcd_mtd0    0.0    30 sleep     0     5
 1264 lockd             0.0    20 sleep     0     0
 1265 nfsd              0.0    20 sleep     0   130
```

```
Stack member 3:

CPU averages:
 1 second: 12%, 20 seconds: 2%, 60 seconds: 2%
System load averages:
 1 minute: 0.03, 5 minutes: 0.02, 15 minutes: 0.00
Current CPU load:
 userspace: 6%, kernel: 4%, interrupts: 1% iowaits: 0%

user processes
==============
  pid name           thrds  cpu%   pri state sleep% runtime
 1544 hostd               1   2.8    20   run     0   120
 1166 exfx               17   1.8    20 sleep     0  3846
 1198 stackd              1   0.9    20 sleep     0   459
 1284 aisexec            44   0.9    -2 sleep     0  2606
    1 init                1   0.0    20 sleep     0   120
 9772 sh                  1   0.0    20 sleep     0     0
 9773 corerotate          1   0.0    20 sleep     0     0
  853 syslog-ng           1   0.0    20 sleep     0   356
  859 klogd               1   0.0    20 sleep     0     1
  910 inetd               1   0.0    20 sleep     0     3
  920 portmap             1   0.0    20 sleep     0     0
  931 crond               1   0.0    20 sleep     0     1
 1090 openhpid           11   0.0    20 sleep     0   233
 1111 hpilogd             1   0.0    20 sleep     0     0
 1240 hsl                 1   0.0    20 sleep     0    79
 1453 authd               1   0.0    20 sleep     0    85
 1477 bgpd                1   0.0    20 sleep     0    40
 1497 cntrd               1   0.0    20 sleep     0     2
 1520 epsrd               1   0.0    20 sleep     0    56
 1571 imi                 1   0.0    20 sleep     0   275
 1594 irdpd               1   0.0    20 sleep     0    23
 1617 lacpd               1   0.0    20 sleep     0    87
 1638 mstpd               1   0.0    20 sleep     0    75
 1662 nsm                 1   0.0    20 sleep     0   163
  1708 pdmd                1   0.0    20 sleep     0    23
 1729 pimd                1   0.0    20 sleep     0    32
 1751 ripd                1   0.0    20 sleep     0    33
 1775 ripngd              1   0.0    20 sleep     0    25
 1797 rmond               1   0.0    20 sleep     0    64
 1963 ntpd                1   0.0    20 sleep     0    15
 2102 atlgetty            1   0.0    20 sleep     0     0
 2712 rpc.statd           1   0.0    20 sleep     0     0
 2716 rpc.statd           1   0.0    20 sleep     0     0
 2722 rpc.mountd          1   0.0    20 sleep     0     0
 2821 automount           1   0.0    20 sleep     0    82
 2892 ntpd                1   0.0    20 sleep     0    17
 2912 sshd                1   0.0    20 sleep     0     0
 9774 login               1   0.0    20 sleep     0     2
12689 more                1   0.0    20 sleep     0     0
```

Table 8-2: Parameters in the output of the **show cpu** command

Parameter	Description
`STACK member`	The stack member output being displayed.
`CPU averages`	Average CPU utilization for the periods stated.
`System load averages`	The average number of processes waiting for CPU time for the periods stated.
`Current CPU load`	Current CPU utilization specified by load types.
`pid`	Identifier number of the process.
`name`	A shortened name for the process
`thrds`	Number of threads in the process.
`cpu%`	Percentage of CPU utilization that this process is consuming.
`pri`	Process priority state.
`state`	Process state; one of "run", "sleep", "zombie", and "dead".
`sleep%`	Percentage of time that the process is in the sleep state.
`runtime`	The time that the process has been running for, measured in jiffies. A jiffy is the duration of one tick of the system timer interrupt.

Related Commands

remote-command <1-4> show
show memory
show memory allocations
show memory history
show memory pools
show process

show cpu history

This command prints a graph showing the historical CPU utilization.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax

```
show [<1-8>]cpu history
```

Mode User Exec and Privileged Exec

Usage This command's output displays three graphs of the percentage CPU utilization:

- per second for the last minute, then
- per minute for the last hour, then
- per 30 minutes for the last 30 hours.

If this command is entered on the stack master, it will print graphs for all the stack members. A stack member heading will be displayed to distinguish the different graphs for every stack member.

If the command is entered on a specific stack member, as a host-directed command, it will print the graph for that particular stack member.

Examples To display a graph showing the historical CPU utilization of the device, use the command:

```
awplus# show cpu history
```

To display the CPU utilization history graph for stack member 3, use the command:

```
awplus# show 3 cpu history
```

where 3 is the node id of the stack member.

Output Figure 8-4: Example output from the **show cpu history** command

```
Stack member 1:

Per second CPU load history

100
 90
 80
 70
 60
 50
 40
 30
 20
 10 ************************************************************
    |....|....|....|....|....|....|....|....|....|....|....|....
    Oldest                                                Newest
        CPU load% per second (last 60 seconds)
              * = average CPU load%


Per minute CPU load history

100        *+
 90        +
 80
 70
 60
 50
 40
 30
 20                        +                 +
 10         ****************************************************
    |....|....|....|....|....|....|....|....|....|....|....|....
    Oldest                                                Newest
        CPU load% per minute (last 60 minutes)
              * = average CPU load%, + = maximum


Per (30) minute CPU load history

100                                                           +
 90
 80
 70
 60
 50
 40
 30
 20
 10                                                           **
    |....|....|....|....|....|....|....|....|....|....|....|....
    Oldest                                                Newest
        CPU load% per 30 minutes (last 60 values / 30 hours)
              * = average, - = minimum, + = maximum
.
.
.
```

Related Commands

remote-command <1-4> show

show memory

show memory allocations

show memory pools

show process

show memory

show memory allocations

show memory pools

show process

show debugging

This command displays information for all debugging options.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax

```
show debugging
```

Default This command runs all the **show debugging** commands in alphabetical order.

Mode User Exec and Privileged Exec

Usage This command displays all debugging information, similar to the way the show tech-support command displays all show output for use by Allied Telesis authorized service personnel only.

Example To display all debugging information, use the command:

```
awplus# show debugging
```

Output Figure 8-5: Example output from the **show debugging** command

```
awplus#show debugging
AAA debugging status:
  Authentication debugging is off
  Accounting debugging is off

802.1X debugging status:

EPSR debugging status:
 EPSR Info debugging is off
 EPSR Message debugging is off
 EPSR Packet debugging is off
 EPSR State debugging is off
IGMP Debugging status:
  IGMP Decoder debugging is off
  IGMP Encoder debugging is off
.
.
.
```

Related Commands

show debugging aaa
show debugging dot1x
show debugging epsr
show debugging igmp
show debugging ip dns forwarding
show debugging lacp
show debugging lldp
show debugging mstp
show debugging pim dense-mode
show debugging pim sparse-mode
show debugging radius
show debugging rip
show debugging snmp
show debugging stack
show debugging vrrp

show ecofriendly

This command displays the switch's eco-friendly configuration status.

Syntax

```
show ecofriendly
```

Mode Privileged Exec

Example To display the switch's eco-friendly configuration status, use the following command:

```
awplus# show ecofriendly
```

Output Figure 8-6: Example output from the **show ecofriendly** command

```
awplus#show ecofriendly
Front panel port LEDs        normal
```

Table 8-3: Parameters in the output of the **show ecofriendly** command

Parameter	Description
`normal`	The eco-friendly feature is disabled and port LEDs show the current state of the ports. This is the default setting.
`off`	The eco-friendly feature is enabled and power to the port LEDs is disabled.

Related Commands ecofriendly led

show interface memory

This command displays the shared memory used by either all interfaces, or the specified interface or interfaces. The output is useful for diagnostic purposes by Allied Telesis authorized service personnel.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax

```
show interface memory
```

```
show interface <port-list> memory
```

Parameter	Description
<port-list>	The ports to display information about. The port list can be: ■ a switch port (e.g. `port1.0.12`) a static channel group (e.g. `sa3`) or a dynamic (LACP) channel group (e.g. `po3`) ■ a continuous range of ports separated by a hyphen, e.g. `port1.0.1-1.0.24`, or `sa1-2`, or `po1-4` ■ a comma-separated list of ports and port ranges, e.g. `port1.0.1,port1.0.4-1.2.24`. Do not mix switch ports, static channel groups, and dynamic (LACP) channel groups in the same list

Mode

User Exec and Privileged Exec

Example

To display the shared memory used by all interfaces, use the command:

```
awplus# show interface memory
```

To display the shared memory used by `port1.0.1` and `port1.0.5 to port1.0.8`, use the command:

```
awplus# show interface port1.0.1,port1.0.5-1.0.8 memory
```

Output

Figure 8-7: Example output from the **show interface <port-list> memory** command

```
awplus#show interface port1.0.1,port1.0.5-1.0.8 memory
Vlan blocking state shared memory usage
---------------------------------------------
Interface     shmid        Bytes Used     nattch     Status
port1.0.1     393228       512            1
port1.0.5     491535       512            1
port1.0.6     557073       512            1
port1.0.7     327690       512            1
port1.0.8     655380       512            1
```

Figure 8-8: Example output from the **show interface memory** command

```
awplus#show interface memory
Vlan blocking state shared memory usage
---------------------------------------------
Interface    shmid        Bytes Used     nattch     Status
port1.0.1    393228       512            1
port1.0.2    458766       512            1
port1.0.3    360459       512            1
port1.0.4    524304       512            1
port1.0.5    491535       512            1
port1.0.6    557073       512            1
port1.0.7    327690       512            1
port1.0.8    655380       512            1
port1.0.9    622611       512            1
.
.
port1.0.21   950301       512            1
port1.0.22   1048608      512            1
port1.0.23   1015839      512            1
port1.0.24   1081377      512            1
lo           425997       512            1
po1          1179684      512            1
po2          1212453      512            1
sa3          1245222      512            1
```

Related Commands show interface brief
show interface status
show interface switchport

show memory

This command displays the memory used by each process that is currently running

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax `show memory [<1-4>] [sort {size|peak|stk}]`

Parameter	Description
`<1-4>`	Specify the stack member number.
`sort`	Changes the sorting order for the list of processes. If you do not specify this, then the list is sorted by percentage memory utilization.
`size`	Sorts the list by the amount of memory the process is currently using.
`peak`	Sorts the list by the peak amount of memory the process has ever used.
`stk`	Sorts the list by the stack size of the process.

Mode User Exec and Privileged Exec

Usage If this command is entered on the stack master, it will display corresponding memory utilization information for all the stack members. A stack member heading will be displayed to distinguish the different lists for every stack member.

If it is entered on a specific stack member, as host-directed commands, it will display corresponding memory utilization information for that stack member.

Example To display the memory used by the current running processes, use the command:

```
awplus# show memory
```

Output Figure 8-9: Example output from the **show memory** command

```
awplus#show memory

Stack member 1:

RAM total: 514920 kB; free: 382716; buffers: 16368 kB

user processes
==============
pid name           mem%   size   peak   data   stk
962 pss               6  33112  36260  27696   244
1   init              0    348   1092    288    84
797 syslog-ng         0    816   2152    752    84
803 klogd             0    184   1244    124    84
843 inetd             0    256   1256    136    84
```

Table 8-4: Parameters in the output of the **show memory** command

Parameter	Description
`STACK member`	The stack member output being displayed.
`RAM total`	Total amount of RAM memory free.
`free`	Available memory size.
`buffers`	Memory allocated kernel buffers.
`pid`	Identifier number for the process.
`name`	Short name used to describe the process.
`mem%`	Percentage of memory utilization the process is currently using.
`size`	Amount of memory currently used by the process.
`peak`	Greatest amount of memory ever used by the process.
`data`	Amount of memory used for data.
`stk`	The stack size.

Related Commands

show memory allocations
show memory history
show memory pools
show memory shared

show memory allocations

This command displays the memory allocations used by processes.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax `show memory allocations [<process>]`

Parameter	Description
<process>	Displays the memory allocation used by the specified process.

Mode User Exec and Privileged Exec

Usage If entered on the stack master, this command will display corresponding memory utilization information for all the stack members. A stack member heading will be displayed to distinguish the different lists for every stack member.

If it is entered on a specific stack member, as host-directed commands, it will display corresponding memory utilization information for that stack member.

Examples To display the memory allocations used by all processes on your device, use the command:

```
awplus# show memory allocations
```

Output Figure 8-10: Example output from the **show memory allocations** command

```
awplus#show memory allocations
Memory allocations for imi
-----------------------------

Current 15093760 (peak 15093760)

Statically allocated memory:
 - binary/exe                  :    1675264
 - libraries                   :    8916992
 - bss/global data             :    2985984
 - stack                       :     139264

Dynamically allocated memory (heap):
 - total allocated             :    1351680
   - in use                    :    1282440
   - non-mmapped               :    1351680
 - maximum total allocated     :    1351680
 - total free space            :      69240
   - releasable                :      68968
   - space in freed fastbins   :         16

Context
           filename:line   allocated      freed
 +            lib.c:749          484
.
.
.
```

Figure 8-11:

Table 8-5: Parameters in the output from the **show memory allocations** command

Parameter	Description
`name`	Short name used to describe the process.
`pid`	Identifier number for the process.
`size`	Amount of memory in kB used by the process.
`peak`	The peak amount of memory in kB ever used by the process.
`data`	Amount of memory used for data.
`stack`	The stack size.

Related Commands

show memory
show memory history
show memory pools
show memory shared
show tech-support

show memory history

This command prints a graph showing the historical memory usage.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax

```
show memory history [<1-4>]
```

Parameter	Description
<1-4>	Specify the stack member number.

Mode User Exec and Privileged Exec

Usage This command's output displays three graphs of the percentage memory utilization:

- per second for the last minute, then
- per minute for the last hour, then
- per 30 minutes for the last 30 hours.

If entered on the stack master, this command will display corresponding memory utilization information for all the stack members. A stack member heading will be displayed to distinguish the different lists for every stack member.

If it is entered on a specific stack member, as host-directed commands, it will display corresponding memory utilization information for that stack member.

Examples To show a graph displaying the historical memory usage for either a single unstacked device, or a complete stack, use the command:

```
awplus# show memory history
```

To show a graph displaying the historical memory usage for a single device (device 3 in this example) within a stack, use the command:

```
awplus# show memory history 3
```

Output Figure 8-12: Example output from the **show memory history** command

```
STACK member 1:

Per minute memory utilization history

100
 90
 80
 70
 60
 50
 40
*************************************************************
 30
 20
 10
    |....|....|....|....|....|....|....|....|....|....|....|....
    Oldest                                                Newest
        Memory utilization% per minute (last 60 minutes)
              * = average memory utilisation%.
.
.
.
----------------------------------------------------------------


Per minute memory utilization history

100
 90
 80
 70
 60
 50
 40
*************************************************************
 30
 20
 10
    |....|....|....|....|....|....|....|....|....|....|....|....
    Oldest                                                Newest
        Memory utilization% per minute (last 60 minutes)
              * = average memory utilisation%.
.
.
.
```

Related Commands
show memory allocations
show memory pools
show memory shared
show tech-support

show memory pools

This command shows the memory pools used by processes.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax `show memory pools [<process>]`

Parameter	Description
<process>	Displays the memory pools used by the specified process.

Mode User Exec and Privileged Exec

Example To shows the memory pools used by processes, use the command:

```
awplus# show memory pools
```

Output Figure 8-13: Example output from the **show memory pools** command

```
awplus#show memory pools
Memory pools for imi
----------------------

Current 15290368 (peak 15290368)

Statically allocated memory:
 - binary/exe                :    1675264
 - libraries                 :    8916992
 - bss/global data           :    2985984
 - stack                     :     139264

Dynamically allocated memory (heap):
 - total allocated           :    1548288
   - in use                  :    1479816
   - non-mmapped             :    1548288
 - maximum total allocated   :    1548288
 - total free space          :      68472
   - releasable              :      68200
   - space in freed fastbins :         16
.
.
.
```

Figure 8-14:

Table 8-6: Parameters in the output from the **show memory pools** command

Parameter	Description
`name`	Short name used to describe the process.
`pid`	Identifier for the process.
`size`	Amount of memory in kB used by the process.
`peak`	Peak amount of memory in kB ever used by the process.
`data`	Amount of memory in kB used for data.
`stack`	The stack size in kB.

Related Commands

show memory allocations
show memory history
show tech-support

show memory shared

This command displays shared memory allocation information. The output is useful for diagnostic purposes by Allied Telesis authorized service personnel.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax `show memory shared`

Mode User Exec and Privileged Exec

Example To display information about the shared memory allocation used on the switch, use the command:

```
awplus# show memory shared
```

Output Figure 8-15: Example output from the **show memory shared** command

```
awplus#show memory shared
Shared Memory Status
--------------------------
Segment allocated  = 39
Pages allocated    = 39
Pages resident     = 11

Shared Memory Limits
--------------------------
Maximum number of segments          = 4096
Maximum segment size (kbytes)       = 32768
Maximum total shared memory (pages) = 2097152
Minimum segment size (bytes)        = 1
```

Related Commands

show memory allocations
show memory history
show memory sort

show process

This command lists a summary of the current running processes.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax

```
show process [<1-4>] [sort {cpu|mem}]
```

Parameter	Description	
<1-4>	The stack member to display output for.	
sort	Changes the sorting order for the list of processes.	
	cpu	Sorts the list by the percentage of CPU utilization.
	mem	Sorts the list by the percentage of memory utilization.

Mode User Exec and Privileged Exec

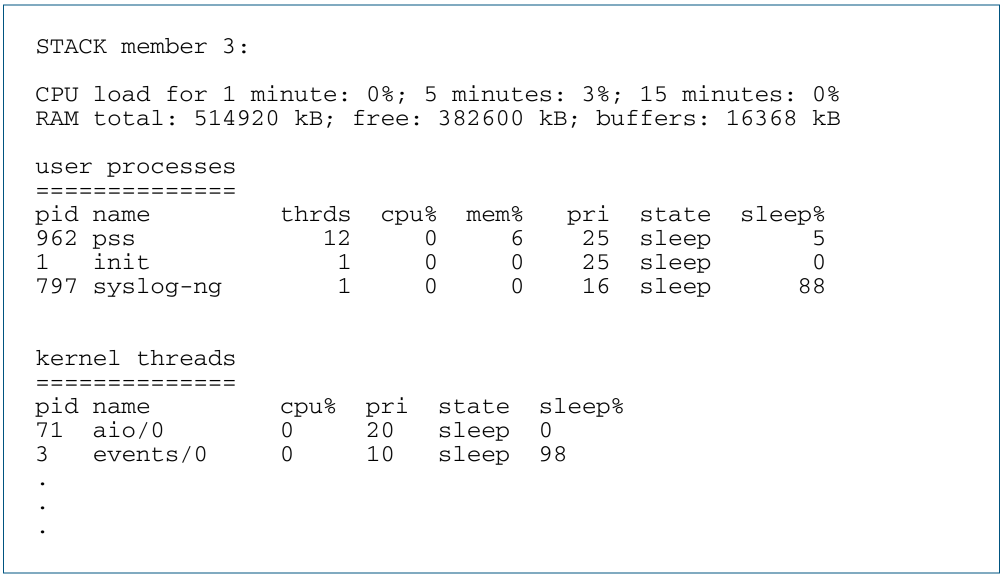
Usage For a stacked configuration, if this command is entered on the stack master, it will display the information for all the stack members. A stack member heading will be displayed to distinguish the different information for every stack member.

If it is entered on a specific stack member, as a host-directed command, it will display the information for that stack member.

Example To display a summary of the current running processes on stack member 3, use the command:

```
awplus# show process 3
```

Output Figure 8-16: Example output from the **show process** command



```
STACK member 3:

CPU load for 1 minute: 0%; 5 minutes: 3%; 15 minutes: 0%
RAM total: 514920 kB; free: 382600 kB; buffers: 16368 kB

user processes
==============
pid name          thrds  cpu%  mem%   pri  state  sleep%
962 pss              12     0     6    25  sleep       5
1   init              1     0     0    25  sleep       0
797 syslog-ng         1     0     0    16  sleep      88


kernel threads
==============
pid name          cpu%  pri  state  sleep%
71  aio/0         0     20   sleep  0
3   events/0      0     10   sleep  98
.
.
.
```

Table 8-7: Parameters in the output from the **show process** command

Parameter	Description
`STACK member`	The stack member output being displayed.
`CPU load`	Average CPU load for the given period.
`RAM total`	Total memory size.
`free`	Available memory.
`buffers`	Memory allocated to kernel buffers.
`pid`	Identifier for the process.
`name`	Short name to describe the process.
`thrds`	Number of threads in the process.
`cpu%`	Percentage of CPU utilization that this process is consuming.
`mem%`	Percentage of memory utilization that this process is consuming.
`pri`	Process priority.
`state`	Process state; one of "run", "sleep", "stop", "zombie", or "dead".
`sleep%`	Percentage of time the process is in the sleep state.

Related Commands

remote-command <1-4> show
show cpu
show cpu history

show reboot history

Use this command to display the switch's reboot history.

Syntax `show reboot history`

Mode User Exec and Privileged Exec

Examples To show the reboot history, use the command:

```
awplus# show reboot history
```

Output Figure 8-17: Example output from the **show reboot history** command

```
------------------------
     Reboot History
------------------------
2010-08-29 20:40:23 (Unexpected) System reboot
2010-08-29 08:26:26 (Unexpected) Rebooting due to critical process (network/nsm) 
failure!
2010-08-29 08:01:56 (Unexpected) System reboot
2010-08-25 22:00:17 (Expected)   CLI(user request)
2010-08-25 20:46:40 (Unexpected) Rebooting due to VCS duplicate member-ID
2010-08-25 20:45:23 (Expected)   CLI(user request)
2010-08-25 20:36:06 (Unexpected) Rebooting due to VCS duplicate master (Continuous 
reboot prevention)
2010-08-25 20:30:50 (Expected)   CLI(user request)
2010-08-25 20:28:04 (Unexpected) System reboot
```

Table 8-8: Parameters in the output from the **show reboot history** command

Parameter	Description
`Unexpected`	Reboot is counted by the continuous reboot prevention feature if the reboot event occurs in the time period specified for continuous reboot prevention.
`Expected`	Reboot is not counted by continuous reboot prevention feature.
`Continuous reboot prevention`	A continuous reboot prevention event has occurred. The action taken is configured with the **continuous-reboot-prevention** command. The next time period during which reboot events are counted begins from this event.
`user request`	User initiated reboot via the CLI.

Related Commands show continuous-reboot-prevention
show tech-support

show router-id

Use this command to show the Router ID of the current system.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax

```
show router-id
```

Mode User Exec and Privileged Exec

Example To display the Router ID of the current system, use the command:

```
awplus# show router-id
```

Output Figure 8-18: Example output from the **show router-id** command

```
awplus>show router-id
Router ID: 10.55.0.2 (automatic)
```

show system

This command displays general system information about the device, including the hardware installed, memory, and software versions loaded. It also displays location and contact details when these have been set.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax show system

Mode User Exec and Privileged Exec

Usage For a stacked configuration, if this command is entered on the stack master, it will display the information for all the stack members. A stack member heading will be displayed to distinguish the different information for every stack member.

If it is entered on a specific stack member, as a host-directed command, it will display the information for that stack member.

Examples To display the system information for a single switch, or a whole stack, use the command:

```
awplus# show system
```

To display the system information of stack member 3, use the command:

```
awplus# remote-command 3 show system
```

Output Figure 8-19: Example output from the **show system** command

```
awplus#show system
Switch System Status                                   Thu Sep 22 14:00:13 2011

Board       ID  Bay   Board Name                         Rev   Serial number
--------------------------------------------------------------------------------
Base       289        x600-24Ts                          X2-0  G1Q67B002
Expansion  306  Bay1  AT-StackXG                         A-0   N/A
--------------------------------------------------------------------------------
RAM:  Total: 513388 kB Free: 419212 kB
Flash: 63.0MB Used: 58.0MB Available: 5.0MB
--------------------------------------------------------------------------------
Environment Status : Normal
Uptime             : 24 days 06:04:58
Bootloader version : 1.1.0-rc12


Current software   : x510-5.4.2A.rel
Software version   : 5.4.2
Build date         : Wed Dec 8 12:13:19 NZDT 2010

Current boot config: flash:/backup.cfg (file exists)
Territory          : usa

System Name
 awplus
System Contact

System Location
```

Figure 8-20: Example output from the **show system** command for a stacked configuration

```
Stack System Status                                     Tue Aug  7 05:25:09 2007

Stack member 1:

Board       ID  Bay   Board Name                             Rev  Serial number
--------------------------------------------------------------------------------
Base       270        x900-24XT                              B-0  41FY68006
Expansion  272  Bay1  XEM-1XP                                B-0  41AR65001
Expansion  285  Bay2  XEM-STK                                A-0  M1L174004
PSU        212  PSU1  AT-PWR01-AC                            F-1  66354904
Fan module 214  PSU2  AT-FAN01                               F-1  66098695
--------------------------------------------------------------------------------
Memory:   DRAM: 514460 kB   Flash: 31.0MB Used: 25.1MB Available: 5.9MB
--------------------------------------------------------------------------------
Environment Status: Normal
Uptime: 0 days 04:26:02


Stack member 2:

Board       ID  Bay   Board Name                             Rev  Serial number
--------------------------------------------------------------------------------
Base       271        x900-24XS                              A-2  41HF6900U
Expansion  272  Bay1  XEM-1XP                                A-0  41AR5B003
Expansion  285  Bay2  XEM-STK                                (nul M1L17400T
PSU        212  PSU1  AT-PWR01-AC                            F-1  66354904
Fan module 214  PSU2  AT-FAN01                               F-1  66098695
--------------------------------------------------------------------------------
Memory:   DRAM: 514460 kB   Flash: 31.0MB Used: 25.2MB Available: 5.8MB
--------------------------------------------------------------------------------
Environment Status: Normal
Uptime: 0 days 00:01:14


Bootloader version : 1.0.9
Current software   : x510-5.4.2A.rel
Software version   : 5.4.2
Build date         : Tue Nov 7 09:27:05 NZST 2011

Current boot config: flash:/default.cfg (file exists)
User Configured Territory: usa

System Name

System Contact

System Location
```

Related Commands remote-command <1-4> show
show system environment

show system environment

This command displays the current environmental status of your device and any attached PSU, XEM, or other expansion option. The environmental status covers information about temperatures, fans, and voltage.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax `show system environment`

Mode User Exec and Privileged Exec

Usage For a stacked configuration, if this command is entered on the stack master, it will display the information for all the stack members. A stack member heading will be displayed to distinguish the different information for every stack member.

If it is entered on a specific stack member, as a host-directed command, it will display the information for that stack member.

Example To display the system's environmental status, use the command:

```
awplus# show system environment
```

Output Figure 8-21: Example output from the **show system environment** command

```
awplus#show system environment
Environment Monitoring Status
Overall Status: Normal

Resource ID: 1  Name: RPS ()
ID  Sensor (Units)                      Reading  Low Limit High Limit Status
1   Primary Power Output                    Yes         -          -      Ok
2   RPS Present                              No         -          -      Ok
3   RPS Power Output                         No         -          -      Ok
4   RPS Fan 1 Good                           No         -          -      Ok
5   RPS Fan 2 Good                           No         -          -      Ok

Resource ID: 2  Name: x600-24Ts
ID  Sensor (Units)                      Reading  Low Limit High Limit Status
1   Fan: Fan 1 (Rpm)                       6888       5000          -     Ok
2   Fan: Fan 2 (Rpm)                       6818       5000          -     Ok
3   Voltage: 2.5V (Volts)                 2.474      2.344      2.865     Ok
4   Voltage: Battery (Volts)              3.150      2.700      3.586     Ok
5   Voltage: 3.3V (Volts)                 3.266      2.973      3.627     Ok
6   Voltage: 5V (Volts)                   5.052      4.505      5.495     Ok
7   Voltage: 12V (Volts)                 11.625     10.813     13.188     Ok
8   Voltage: 1.25V (Volts)                1.223      1.125      1.378     Ok
9   Temp: Internal (Degrees C)               31   48(Hyst)         50     Ok
10  Fan: Fan 3 (Rpm)                       6750       5000          -     Ok
11  Fan: Fan 4 (Rpm)                       6683       5000          -     Ok
12  Voltage: 2.5V (Volts)                 2.474      2.344      2.865     Ok
13  Voltage: 1.2V (Volts)                 1.181      1.083      1.322     Ok
14  Voltage: 3.3V (Volts)                 3.266      2.973      3.627     Ok
15  Voltage: 5V (Volts)                   5.026      4.505      5.495     Ok
16  Voltage: 12V (Volts)                 11.563     10.813     13.188     Ok
17  Voltage: 1.8V (Volts)                 1.772      1.617      1.983     Ok
18  Temp: Internal (Degrees C)               29   43(Hyst)         45     Ok
```

Related Commands show system

show system interrupts

Use this command to display the number of interrupts for each IRQ (Interrupt Request) used to interrupt input lines on a PIC (Programmable Interrupt Controller) on your switch.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax `show system interrupts`

Mode User Exec and Privileged Exec

Example To display information about the number of interrupts for each IRQ in your device, use the command:

```
awplus# show system interrupts
```

Output Figure 8-22: Example output from the **show system interrupts** command

```
awplus>show system interrupts
           CPU0
  1:          2    CPM2 SIU  Level Enabled   0     i2c-mpc
  2:        145    CPM2 SIU  Level Enabled   0     spi-mpc
 77:          0    OpenPIC   Level Enabled   0     enet_tx
 78:          2    OpenPIC   Level Enabled   0     enet_rx
 82:          0    OpenPIC   Level Enabled   0     enet_error
 90:       5849    OpenPIC   Level Enabled   0     serial
 91:    2066672    OpenPIC   Level Enabled   0     i2c-mpc
 94:        147    OpenPIC   Level Enabled   0     cpm2_cascade
112:          5    OpenPIC   Edge  Enabled   0     phy_interrupt
114:     398714    OpenPIC   Level Enabled   0     mvPP
115:      26247    OpenPIC   Level Enabled   0     mvPP
119:          0    OpenPIC   Edge  Enabled   0     Power supply status
BAD:          0
```

Related Commands show system environment

show system pci device

Use this command to display the PCI devices on your switch.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax

```
show system pci device Mode
```

User Exec and Privileged Exec

Example

To display information about the PCI devices on your switch, use the command:

```
awplus# show system pci device
```

Output

Figure 8-23: Example output from the **show system pci device** command

```
awplus>show system pci device
00:0c.0 Class 0200: 11ab:00d1 (rev 01)
        Flags: bus master, 66Mhz, medium devsel, latency 128, IRQ 113
        Memory at 5ffff000 (32-bit, non-prefetchable) [size=4K]
        Memory at 58000000 (32-bit, non-prefetchable) [size=64M]

00:0d.0 Class 0200: 11ab:00d1 (rev 01)
        Flags: bus master, 66Mhz, medium devsel, latency 128, IRQ 116
        Memory at 57fff000 (32-bit, non-prefetchable) [size=4K]
        Memory at 50000000 (32-bit, non-prefetchable) [size=64M]
```

Related Commands

show system environment
show system pci tree

show system pci tree

Use this command to display the PCI tree on your switch.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax

```
show system pci tree
```

Mode

User Exec and Privileged Exec

Example

To display information about the PCI tree on your switch, use the command:

```
awplus# show system pci tree
```

Output

Figure 8-24: Example output from the **show system pci tree** command

```
awplus>show system pci tree
-[00]-+-0c.0  11ab:00d1
      \-0d.0  11ab:00d1
```

Related Commands

show system environment
show system pci device

show system pluggable

This command displays brief pluggable transceiver information showing the pluggable type, the pluggable serial number, and the pluggable port on the switch. Different types of pluggable transceivers, such as SFPs, are supported in different models of switch. See your Allied Telesis dealer for more information about the models of pluggables that your switch supports.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax

```
show system pluggable [<port-list>]
```

Parameter	Description
<port-list>	The ports to display information about. The port list can be: ■ a switch port (e.g. `port1.0.12`) ■ a continuous range of ports separated by a hyphen, e.g. `port1.0.1-1.0.24` ■ a comma-separated list of ports and port ranges, e.g. `port1.0.1,port1.0.4-1.2.24`.

Mode User Exec and Privileged Exec

Usage For a stacked configuration, if this command is entered on the stack master, it will display information about the pluggable transceivers for all the stack members. A stack member heading will be displayed to distinguish different pluggable transceiver information for every stack member.

If it is entered on a specific stack member, as a host-directed command, it will display information about the pluggable transceiver for that stack member.

Example To display brief information about pluggable transceivers installed in `port1.0.21` through `port1.0.24`, use the command:

```
awplus# show system pluggable port1.0.21-1.0.24
```

Output Figure 8-25: Example output from the **show system pluggable port1.0.21-1.0.24** command

```
System Pluggable Information
Port   Manufacturer     Device            Serial Number    Datecode Type
--------------------------------------------------------------------------------
1.0.21 AGILENT          HFBR-5710L        0401312315461272 040131   1000BASE-SX
1.0.22 AGILENT          QBCU-5730R        AK0614GKF7       060408   1000BASE-T
1.0.23 AGILENT          HFBR-5710L        0305130112182696 030513   1000BASE-SX
1.0.24 AGILENT          HBCU-5710R        AK051300SM       050402   1000BASE-T
--------------------------------------------------------------------------------
```

Example To display information about the pluggable transceiver installed in `port1.0.21`, use the command:

```
awplus# show system pluggable port1.0.21
```

Output Figure 8-26: Example output from the **show system pluggable port1.0.21** command

```
System Pluggable Information
Port   Manufacturer     Device           Serial Number    Datecode Type
--------------------------------------------------------------------------------
1.0.21 AGILENT          HFBR-5710L       0401312315461272 040131   1000BASE-SX
--------------------------------------------------------------------------------
```

Table 8-9: Parameters in the output from the **show system pluggables** command

Parameter	Description
`Port`	Specifies the vendor's name for the installed XFP or SFP.
`Manufacturer`	Specifies the device name for the installed XFP or SFP.
`Device`	Specifies the device type for the installed XFP or SFP, such as 1000BASE-LX for an SFP.
`Serial Number`	Specifies the serial number for the installed SFP.
`Datecode`	Specifies the manufacturing datecode for the installed SFP. Checking the manufacturing datecode with the vendor may be useful when determining Laser Diode aging issues.
`Type`	Specifies the laser wavelength of the installed pluggable transceiver.

Related Commands show system environment
show system pluggable detail
show system pluggable diagnostics

show system pluggable detail

This command displays detailed pluggable transceiver information showing the pluggable type, the pluggable serial number, and the pluggable port on the switch. Different types of pluggable transceivers, such as SFPs or XFPs, are supported in different models of switch. See your Allied Telesis dealer for more information about the models of pluggables that your switch supports.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax

```
show system pluggable [<port-list>] detail
```

Parameter	Description
<port-list>	The ports to display information about. The port list can be: ■ a switch port (e.g. `port1.0.12`) ■ a continuous range of ports separated by a hyphen, e.g. `port1.0.1-1.0.24` ■ a comma-separated list of ports and port ranges, e.g. `port1.0.1,port1.0.4-1.2.24`.

Mode User Exec and Privileged Exec

Usage For a stacked configuration, if this command is entered on the stack master, it will display detailed information about the pluggable transceivers for all the stack members. A stack member heading will be displayed to distinguish the different pluggable transceiver information for every stack member.

If it is entered on a specific stack member, as a host-directed command, it will display detailed information about the pluggable transceiver for that stack member.

In addition to the information about pluggable transceivers displayed using the show system pluggable command (port, manufacturer, serial number, manufacturing datecode, and type information), the **show system pluggable detail** command displays the following information:

- **SFP Laser Wavelength**: Specifies the laser wavelength of the installed pluggable transceiver
- **Single mode Fiber**: Specifies the link length supported by the pluggable transceiver using single mode fiber
- **OM1 (62.5µm) Fiber**: Specifies the link length (in µm - micron) supported by the pluggable transceiver using 62.5 micron multi-mode fiber.
- **OM2 (50 µm) Fiber**: Specifies the link length (in µm - micron) supported by the pluggable transceiver using 50 micron multi-mode fiber.
- **Diagnostic Calibration**: Specifies whether the SFP pluggable supports DDM or DOM Internal or External Calibration.
  - « **Internal** is displayed if the SFP pluggable supports DDM or DOM Internal Calibration.
  - « **External** is displayed if the SFP pluggable support DDM or DOM External Calibration.
  - « **-** is displayed if SFP DDM Internal Calibration or External Calibration is not supported.
- **Power Monitoring**: Displays the received power measurement type, which can be either **OMA** (Optical Module Amplitude) or **Avg** (Average Power) measured in µW.

- **FEC BER support**: Specifies whether FEC (Forward Error Correction) coder can generate a BER (Bit Error Rate) signal that is used as feedback to tune an XFP if DOM is supported.

Note For parameters that are not supported or not specified, a hyphen is displayed instead. FEC BER support may be available on an XFP if the XFP supports DOM. FEC BER support is not available on an SFP or an SFP+ even if the SFP or the SFP+ supports DDM. FEC BER support is applicable to XFPs that support DOM only.

Example To display detailed information about the pluggable transceivers installed on a standalone switch, use the command:

```
awplus# show system pluggable port1.0.24 detail
```

Output Figure 8-27: Example output from the **show system pluggable detail** command on a switch

```
awplus#show system pluggable port1.0.24 detail
System Pluggable Information Detail

Port1.0.24
==========
Vendor Name:                AGILENT
Device Name:                HFCT-5710L
Device Type:                1000BASE-LX
Serial Number:              0402142241184360
Manufacturing Datecode:     040214
SFP Laser Wavelength:       -
Link Length Supported
 Single Mode Fiber :       10Km
 OM1 (62.5um) Fiber:       550m
 OM2 (50um) Fiber  :       550m
Diagnostic Calibration:     Internal
Power Monitoring:           Avg
FEC BER support:            -
```

Table 8-10: Parameters in the output from the **show system pluggables detail** command:

Parameter	Description
`Stack member`	The stack member output being displayed.
`Port`	Specifies the port the SFP is installed in.
`Vendor Name:`	Specifies the vendor's name for the installed SFP.
`Device Name:`	Specifies the device name for the installed SFP.
`Device Type:`	Specifies the device type for the installed SFP.
`Serial Number:`	Specifies the serial number for the installed SFP.
`Manufacturing Datecode:`	Specifies the manufacturing datecode for the installed SFP. Checking the manufacturing datecode with the vendor may be useful when determining Laser Diode aging issues.
`SFP Laser Wavelength:`	Specifies the laser wavelength of the installed pluggable transceiver.
`Single Mode Fiber:`	Specifies the link length supported by the pluggable transceiver using single mode fiber.
`OM1 (62.5um) Fiber:`	Specifies the link length (in µm - micron) supported by the pluggable transceiver using 62.5 micron multi-mode fiber.
`OM2 (50um) Fiber:`	Specifies the link length (in µm - micron) supported by the pluggable transceiver using 50 micron multi-mode fiber.
`Diagnostic Calibration:`	Specifies whether the SFP pluggable supports DDM Internal or External Calibration: **Internal** is displayed if the SFP pluggable supports DDM Internal Calibration. **External** is displayed if the SFP pluggable support DDM External Calibration. **-** is displayed if SFP DDM Internal Calibration or External Calibration is not supported.
`Power Monitoring:`	Displays the received power measurement type, which can be either **OMA** (Optical Module Amplitude) or **Avg** (Average Power) measured in µW.
`FEC BER support:`	Specifies whether FEC (Forward Error Correction) coder can generate a BER (Bit Error Rate) signal that is used as feedback to tune an XFP if DOM is supported.

Related Commands show system environment
show system pluggable
show system pluggable diagnostics

show system pluggable diagnostics

This command displays diagnostic information about pluggable transceivers, such as SFPs and XFPs, which support Digital Diagnostic Monitoring (DDM) for SFPs or Digital Optical Monitoring (DOM) for XFPs that are installed in your switch. Different types of pluggable transceivers, such as SFPs or XFPs, are supported in different models of switch. See your Allied Telesis dealer for more information about the models of pluggables that your switch supports.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax

```
show system pluggable [<port-list>] diagnostics
```

Parameter	Description
<port-list>	The ports to display information about. The port list can be: ■ a switch port (e.g. `port1.0.12`) ■ a continuous range of ports separated by a hyphen, e.g. `port1.0.1-1.0.24` ■ a comma-separated list of ports and port ranges, e.g. `port1.0.1,port1.0.4-1.2.24`.

Mode User Exec and Privileged Exec

Usage For a stacked configuration, if this command is entered on the stack master, it will display information about the pluggable transceivers for all the stack members. A stack member heading will be displayed to distinguish different pluggable transceiver information for every stack member.

If it is entered on a specific stack member, as a host-directed command, it will display information about the pluggable transceiver for that stack member.

Modern optical SFP transceivers support Digital Diagnostics Monitoring (DDM) functions. Modern optical XFP transceivers support Digital Optical Monitoring (DOM) functions.

Diagnostic monitoring features allow you to monitor real-time parameters of the SFP or XFP, such as optical output power, optical input power, temperature, laser bias current, and transceiver supply voltage. Additionally, RX LOS (Loss of Signal) is shown when the received optical level is below a preset threshold. Monitoring these parameters to check on the health of all SFP and XFP transceivers, selected SFP and XFP transceivers or a specific SFP or XFP transceiver installed in a switch.

Example To display detailed information about all pluggable transceivers installed on a standalone switch, use the command:

```
awplus# show system pluggable diagnostics
```

Output Figure 8-28: Example output from the **show system pluggable diagnostics** command on a switch

```
awplus#show system pluggable diagnostics
System Pluggable Information Diagnostics

Port1.0.21          Status             Alarms                   Warnings
Reading   Alarm     Max     Min   Warning     Max     Min
Temp: (Degrees C)  29.387       -  100.00  -40.00         -  85.000  -10.00
Vcc: (Volts)        3.339       -   3.465   3.135         -   3.400   3.200
Tx Bias: (mA)      10.192       -  37.020   3.260         -  34.520   5.760
Tx Power: (mW)     17.872       -  35.643   8.953         -  28.313  11.271
Rx Power: (mW)      0.006     Low  15.849   0.025       Low  12.589   0.040
Rx LOS:           Rx Down

Port1.0.22          Status             Alarms                   Warnings
Reading   Alarm     Max     Min   Warning     Max     Min
Temp: (Degrees C)  29.387       -  100.00  -40.00         -  85.000  -10.00
Vcc: (Volts)        3.378       -   3.630   2.970         -   3.465   3.135
Tx Bias: (mA)       2.802       -   6.000   1.000         -   5.000   1.000
Tx Power: (mW)      2.900       -  11.000   0.600         -  10.000   0.850
Rx Power: (mW)      1.739       -  18.000   0.000         -  10.000   0.200
Rx LOS:             Rx Up
```

Example To display detailed information about the pluggable transceiver installed in `port1.0.22` on a standalone switch, use the command:

```
awplus# show system pluggable diagnostics port1.0.22
```

Output Figure 8-29: Example output from the **show system pluggable diagnostics port1.0.22** command on a switch

```
awplus#show system pluggable port1.0.22 diagnostics
System Pluggable Information Diagnostics

Port1.0.22          Status             Alarms                   Warnings
Reading   Alarm     Max     Min   Warning     Max     Min
Temp: (Degrees C)  29.387       -  100.00  -40.00         -  85.000  -10.00
Vcc: (Volts)        3.378       -   3.630   2.970         -   3.465   3.135
Tx Bias: (mA)       2.802       -   6.000   1.000         -   5.000   1.000
Tx Power: (mW)      2.900       -  11.000   0.600         -  10.000   0.850
Rx Power: (mW)      1.739       -  18.000   0.000         -  10.000   0.200
Rx LOS:             Rx Up
```

Table 8-11: Parameters in the output from the **show system pluggables diagnostics** command:

Parameter	Description
`Temp: (Degrees C)`	Shows the temperature inside the XFP, SFP or SFP+ transceiver.
`Vcc: (Volts)`	Shows voltage supplied to the XFP, SFP or SFP+ transceiver.
`Tx Bias: (mA)`	Shows current to the Laser Diode in the XFP, SFP or SFP+ transceiver.
`Tx Power: (mW)`	Shows the amount of light transmitted from the XFP, SFP or SFP+ transceiver.
`Rx Power: (mW)`	Shows the amount of light received in the XFP, SFP or SFP+ transceiver.
`Rx LOS:`	Shows when the received optical level falls below a preset threshold.

Related Commands

show system environment
show system pluggable
show system pluggable detail

show system serialnumber

This command shows the serial number information for the switch.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax `show system serialnumber`

Mode User Exec and Privileged Exec

Example To display the serial number information for the switch, use the command:

```
awplus# show system serialnumber
```

Output Figure 8-30: Example output from the **show system serialnumber** command

```
awplus#show system serialnumber
45AX5300X
```

show tech-support

The **show tech-support** command generates system and debugging information for the switch and saves it to a file. You can optionally limit it to display only information for a given protocol.

The command generates a large amount of output and the output is saved into a file. The output file name can be specified by the **outfile** option. If the output file already exists, a new file name is generated with the current time stamp. Since output files may be too large for Flash on the switch we recommend saving files to a USB storage device whenever possible to avoid switch lockup.

If **all** is specified the command captures the full list of information of the device. If **system** is specified the command captures general system information of the device.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax

```
show tech-support [all] [outfile <filename>]

show tech-support {[dhcpsn] [epsr] [ip] [ipv6]  [rinpng] [mld] [stp]
    [system]}
    [outfile <filename>]
```

Parameter	Description
`all`	Output full troubleshooting information for all protocols and the device.
`dhcpsn`	Output only DHCP snooping specific troubleshooting information.
`epsr`	Output only EPSR protocol specific troubleshooting information.
`igmp`	Output only IGMP protocol specific troubleshooting information.
`ip`	Output only IP protocol specific troubleshooting information.
`ipv6`	Output only IPv6 protocol specific troubleshooting information.
`pim`	Output only PIM protocol specific troubleshooting information.
`mld`	Output only MLD protocol specific troubleshooting information.
`stack`	Output only stacking device specific troubleshooting information.
`stp`	Output only STP protocol specific troubleshooting information.
`system`	Output general system (not protocol) troubleshooting information.
`outfile`	Keyword used to specify the file name for the output file.
`<filename>`	Placeholder used to specify the file name for the output file.

Default The **show tech-support** command by default captures **all** information for the switch.

By default the output is saved to the file '`tech-support.txt.gz`' in the current directory. If this file already exists in the current directory then a new file is generated with the time stamp appended to the file name, for example '`tech-support20080109.txt.gz`', so the last saved file is retained.

Mode Privileged Exec

Usage The **show tech-support** command is useful for collecting a large amount of information about all protocols or specific protocols on your switch for troubleshooting purposes. The output of this command can be provided to technical support representatives when reporting a problem.

Examples To capture the full set of show output for the technical support, use the command:

```
awplus# show tech-support
```

To capture show output related to IP module and save it to a file named `support-ip.txt.gz` a USB storage device, use the following command:

```
awplus# show tech-support ip outfile usb:support-ip.txt.gz
```

To capture the system technical support information, use the below command:

```
awplus# show tech-support system
```

Output The output of this command may include the result of the following commands:

show arp
show boot
show counter dhcp-client
show counter dhcp-relay
show counter dhcp-server
show counter log
show counter mail
show counter ntp
show counter ping-poll
show counter snmp-server
show cpu
show cpu history
show diagnostic channel-group
show etherchannel
show etherchannel detail
show exception log
show interface
show interface brief
show ip igmp groups
show ip igmp interface
show ip igmp snooping mrouter vlan1 (see the show ip igmp snooping mrouter command)
show ip interface
show ip pim sparse-mode bsr-router
show ip pim sparse-mode interface detail
show ip pim sparse-mode mroute detail
show ip pim sparse-mode neighbor
show ip pim sparse-mode nexthop
show ip pim sparse-mode rp mapping
show ip route
show lacp-counter
show lacp sys-id
show license
show log
show log permanent
show memory
show memory allocations
show memory history
show memory pools

show ntp associations
show ntp status
show platform
show platform port
show reboot history
show running-config
show spanning-tree
show startup-config
show static-channel-group
show system
show system environment
show users
show vlan brief (see the show vlan command)
show vrrp

speed (asyn)

This command changes the console speed from the switch. Note that a change in console speed is applied for subsequent console sessions. Exit the current session to enable the console speed change using the clear line console command.

Syntax `speed <console-speed-in-bps>`

Parameter	Description
<console-speed-in-bps>	Console speed Baud rate in bps (bits per second).
`1200`	1200 Baud
`1800`	1800 Baud
`2400`	2400 Baud
`9600`	9600 Baud
`19200`	19200 Baud
`38400`	38400 Baud
`57600`	57600 Baud
`115200`	115200 Baud

Default The default console speed baud rate is 9600 bps.

Mode Line Configuration

Usage This command is used to change the console (asyn) port speed. Set the console speed to match the transmission rate of the device connected to the console (asyn) port on your switch.

Example To set the terminal console (asyn0) port speed from the switch to 115200 bps, then exit the session, and log in again to enable the change, use the commands:

```
           awplus# configure terminal
   awplus(config)# line console 0
awplus(config-line)# speed 115200
awplus(config-line)# exit
   awplus(config)# exit
           awplus# exit
```

The new console speed of 115200 bps is applied after exiting the session and before login.

```
awplus login:
    Password:

      awplus>
```

Related Commands line
clear line console
show running-config
show startup-config
speed

system territory

This command sets the territory of the system.

Use the **no** variant of this command to return the territory to its default setting of `japan`.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax

```
system territory {australia|nz|europe|japan|usa|china|korea}
```

```
no system territory
```

Parameter	Description
`australia`	Australia
`nz`	New Zealand
`europe`	Europe
`japan`	Japan
`usa`	USA
`china`	China
`korea`	Korea

Mode Global Configuration

Example To set the territory to USA, enter the command:

```
awplus(config)# system territory usa
```

Validation Commands show system

terminal monitor

Use this command to display debugging output on a terminal.

To display the cursor after a line of debugging output, press the Enter key.

Use the command **terminal no monitor** to stop displaying debugging output on the terminal, or use the timeout option to stop displaying debugging output on the terminal after a set time.

Syntax

```
terminal monitor [<1-60>]
```

```
terminal no monitor
```

Parameter	Description
<1-60>	Set a timeout between 1 and 60 seconds for terminal output.

Default Disabled

Mode Privileged Exec

Examples To display debugging output on a terminal, enter the command:

```
awplus# terminal monitor
```

To specify timeout of debugging output after 60 seconds, enter the command:

```
awplus# terminal monitor 60
```

To stop displaying debugging output on the terminal, use the command:

```
awplus# terminal no monitor
```

Related Commands All debug commands

undebug all

This command applies the functionality of the no debug all command.

Chapter 9: Debugging and Logging

Introduction

AlliedWare Plus^{TM} has a comprehensive debugging and logging facility in various protocols and components. This chapter describes how to start/stop debugging and logging. For detailed descriptions of the commands used to configure logging, see Chapter 10, Logging Commands.

Debugging

Many protocols have debug commands. Debug commands, when used with the parameters, log protocol-specific information. For example, using the **debug mstp protocol** command, results in the device writing all debugging messages generated by the MSTP algorithm to the logging system.

On using a debug command, the protocol continues to generate output until the **no** parameter is used with the command. To specify where logging output is sent, and the level of events to log, use the **log** commands in Chapter 10, Logging Commands.

Logging to terminal

To start debugging to the terminal:

Step 1: Turn on the debug options by using the relevant debug command.

Step 2: Run the terminal monitor command.

```
awplus> enable
awplus# configure terminal
awplus(config)# debug <protocol> (parameter)
awplus(config)# exit
awplus# terminal monitor
```

Sample Output This is a sample output of the **debug rsvp events** command displayed on the terminal:

```
awplus#terminal monitor

Dec  2 16:41:49 localhost RSVP[6518]: RSVP: RSVP message sent to
10.10.23.60/32 via interface vlan2

Dec  2 16:41:57 localhost RSVP[6518]: RSVP: Received an RSVP message
of type RSVP Reservation from 192.168.0.60 via interface vlan2

Dec  2 16:41:57 localhost RSVP[6518]: RSVP: Received a RESV message
from 10.10.23.60/32
```

Turning off debugging

To turn off debugging, use the `no debug` or `undebug` command. When a protocol is specified with the `no debug` or `undebug` commands, debugging is stopped for the specified protocol. To stop all debugging, use the `all` parameter with these commands.

```
awplus#undebug all
```

Logging

Protocols generate important debugging messages by default, and send them to the logging system. Additional more detailed messages can be generated by enabling debugging ("Debugging" on page 9.2).

Messages can be filtered based on: the program that generated the message, the severity level of the message, the type of facility that generated the message, substrings within the message text. The severity levels in order are:

- emergencies
- alerts
- critical
- errors
- warnings
- notifications
- informational
- debugging

The facility categories are:

- auth Security/authorization messages
- authpriv Security/authorization messages (private)
- cron Clock daemon
- daemon System daemons
- ftp FTP daemon
- kern Kernel messages
- lpr Line printer subsystem
- mail Mail system
- news Network news subsystem
- syslog Messages generated internally by syslogd
- user Random user-level messages
- uucp UUCP subsystem

Log Outputs

The following types of logging output are available:

- buffered
- permanent
- terminal
- console
- host
- email

Buffered log

The buffered log is a file stored in RAM on the device. Because it is stored in RAM its content does not survive a reboot of the device. A device can only have one instance of the buffered log. The buffered log is enabled by default and has a filter to include messages with a severity level of 'notifications' and above. The buffered log can be enabled or disabled using the commands:

```
        awplus# configure terminal
awplus(config)# log buffered
awplus(config)# no log buffered
```

Additional filters can be added and removed using the commands described in log buffered (filter) command on page 10.9:

```
awplus(config)# log buffered {facility|level|msgtext|program}
awplus(config)# no log buffered {facility|level|msgtext|
                program}
```

The following log buffered commands are available:

`show log`	Displays the entire contents of the buffered log
`show log tail`	Displays the 10 most recent entries in the buffered log.
`show log tail <10-250>`	Displays a specified number of the most recent entries in the buffered log.
`show log config`	Displays the configuration of all log outputs
`log buffered size`	Specify the amount of memory the buffered log may use.
`clear log`	Remove the contents of the buffered log (and permanent log if it exists)
`clear log buffered`	Remove the contents of the buffered log only
`default log buffered`	Restore the buffered log to its default configuration

Permanent log

The permanent log is a file stored in NVS on the device. This output type is only available on devices that have NVS. The contents on the permanent log is retained over a reboot. A device can only have one instance of the permanent log. The permanent log is enabled by default and has a filter to include messages with a severity level of 'warning' and above. The permanent log can be disabled using the command:

```
        awplus# configure terminal
awplus(config)# no log permanent
```

Additional filters can be added and removed using the commands described in log permanent (filter):

```
        awplus# configure terminal
awplus(config)# log permanent {facility|level|msgtext|
                program}
awplus(config)# no log permanent {facility|level|msgtext|
                program}
```

Table 9-1: Permanent log commands

Command	Description
`show log permanent`	Display the entire contents of the permanent log
`show log permanent tail`	Display the 10 most recent entries in the permanent log
`show log permanent tail <10-250>`	Display a specified number of the most recent entries in the permanent log
`show log config`	Display the configuration of all log outputs
`log permanent size`	Specify the amount of memory the permanent log may use
`clear log`	Remove the contents of the buffered log and permanent log
`clear log permanent`	Remove the contents of the permanent log only
`default log permanent`	Restore the permanent log to its default configuration

Host log

A host log sends log messages to a remote syslog server. A device may have many syslog hosts configured. To configure or remove a host use the commands:

```
        awplus# configure terminal
awplus(config)# log host <ip-addr>9
awplus(config)# no log host <ip-addr>9
```

where `<ip-addr>` is the IP address of the remote syslog server.

There are no default filters associated with host outputs when they are created. Filters can be added and removed with the log host (filter) command on page 10.23.

It is not possible to view the log messages sent to this type of output as they are not retained on the device. They must be viewed on the remote device. The other host log commands are:

show log config	Displays the configuration of all log outputs
log host time	Adjust the time information in messages to a time zone other than the one configured on this device
default log host <ip-address>	Restores the device default settings for log sent to a remote syslog server.

Email log

An email log sends log messages to an email address. A device may have many email logs configured. To configure or remove an email log use the commands:

```
        awplus#  configure terminal
awplus(config)#  log email <email-address>
awplus(config)#  no log email <email-address>
```

where <email-address> is the destination email address.

There are no default filters associated with email outputs when they are created. Filters can be added and removed with the commands described in log email (filter):

```
        awplus#  configure terminal
awplus(config)#  log email <email-address> {facility|level|
                 msgtext|program}
awplus(config)#  no log email <email-address> {facility|
                 level|msgtext|program}
```

It is not possible to view the log messages sent to this type of output as they are not retained on the device. They must be viewed by the email recipient.

The other email log commands are:

show log config	Displays the configuration of all log outputs
log email time	Adjust the time information in messages to a time zone other than the one configured on this device
default log email	Restores the device default settings for log messages sent to an email address.

Note An email server and "from" address must be configured on the device in order for email logs to work:

- mail from <*email-address*>
- mail smtpserver <*ip-address*>

where the <*email-address*> is the 'From:' field on the sent email, and the <*ip-address*> is the email's destination SMTP server.

Email logs are sent in batches of approximately 20 messages and have the subject line "Log messages"

Chapter 10: Logging Commands

Command List

This chapter provides an alphabetical reference of commands used to configure logging.

clear exception log

This command resets the contents of the exception log, but does not remove the associated core files.

Note When this command is used within a virtual chassis stack (VCS), it will remove the contents of the exception logs in all stack members.

Syntax

```
clear exception log
```

Mode Privileged Exec

Example

```
awplus# clear exception log
```

clear log

This command removes the contents of the buffered and permanent logs.

Note When this command is used within a virtual chassis stack (VCS), it will remove the contents of the buffered and permanent logs in all stack members.

Syntax

```
clear log
```

Mode Privileged Exec

Example To delete the contents of the buffered and permanent log use the command:

```
awplus# clear log
```

Validation Commands show log

Related Commands clear log buffered
clear log permanent

clear log buffered

This command removes the contents of the buffered log.

Note When this command is used within a virtual chassis stack (VCS), it will remove the contents of the buffered and permanent logs in all stack members.

Syntax

```
clear log buffered
```

Mode Privileged Exec

Example To delete the contents of the buffered log use the following commands:

```
awplus# clear log buffered
```

Validation Commands show log

Related Commands clear log
clear log permanent

clear log permanent

This command removes the contents of the permanent log.

Note When this command is used within a virtual chassis stack (VCS), it will remove the contents of the buffered and permanent logs in all stack members.

Syntax

```
clear log permanent
```

Mode Privileged Exec

Example To delete the contents of the permanent log use the following commands:

```
awplus# clear log permanent
```

Validation Commands show log

Related Commands clear log
clear log buffered

default log buffered

This command restores the default settings for the buffered log stored in RAM. By default the size of the buffered log is 50 kB and it accepts messages with the severity level of "warnings" and above.

Syntax `default log buffered`

Default The buffered log is enabled by default.

Mode Global Configuration

Example To restore the buffered log to its default settings use the following commands:

```
        awplus# configure terminal
awplus(config)# default log buffered
```

Validation Commands show log config

Related Commands log buffered
log buffered size

default log console

This command restores the default settings for log messages sent to the terminal when a log console command is issued. By default all messages are sent to the console when a **log console** command is issued.

Syntax `default log console`

Mode Global Configuration

Example To restore the log console to its default settings use the following commands:

```
        awplus# configure terminal
awplus(config)# default log console
```

Validation Commands show log config

Related Commands log console
log console (filter)

default log email

This command restores the default settings for log messages sent to an email address. By default no filters are defined for email addresses. Filters must be defined before messages will be sent. This command also restores the remote syslog server time offset value to local (no offset).

Syntax

```
default log email <email-address>
```

Parameter	Description
<email-address>	The email address to send log messages to

Mode Global Configuration

Example To restore the default settings for log messages sent to the email address admin@alliedtelesis.com use the following commands:

```
        awplus# configure terminal
awplus(config)# default log email admin@alliedtelesis.com
```

Related Commands show log config

default log host

This command restores the default settings for log sent to a remote syslog server. By default no filters are defined for remote syslog servers. Filters must be defined before messages will be sent. This command also restores the remote syslog server time offset value to local (no offset).

Syntax

```
default log host <ip-addr>
```

Parameter	Description
<ip-addr>	The IP address of a remote syslog server

Mode Global Configuration

Example To restore the default settings for messages sent to the remote syslog server with IP address 10.32.16.21 use the following commands:

```
        awplus# configure terminal
awplus(config)# default log host 10.32.16.21
```

Validation Commands show log config

Related Commands log email

default log monitor

This command restores the default settings for log messages sent to the terminal when a terminal monitor command is used.

Syntax

```
default log monitor
```

Default All messages are sent to the terminal when a terminal monitor command is used.

Mode Global Configuration

Example To restore the log monitor to its default settings use the following commands:

```
        awplus# configure terminal
awplus(config)# default log monitor
```

Related Commands

log monitor (filter)
show log config

default log permanent

This command restores the default settings for the permanent log stored in NVS. By default, the size of the permanent log is 50 kB and it accepts messages with the severity level of `warnings` and above.

Syntax

```
default log permanent
```

Default The permanent log is enabled by default.

Mode Global Configuration

Example To restore the permanent log to its default settings use the following commands:

```
        awplus# configure terminal
awplus(config)# default log permanent
```

Related Commands

log permanent
log permanent size
show log config

exception coredump size

This command sets the size of core files, and can also be used to stop core files being created.

Use the **no** variant of this command to restore the core file size to its default (unlimited).

This setting only applies to processes created after this command has been executed, to ensure this is applied to all processes the system will need to be restarted.

Syntax

```
exception coredump size {none|small|medium|large|unlimited}
```

```
no exception coredump size
```

Parameter	Description
`none`	Don't create corefiles
`small`	Create small corefiles
`medium`	Create medium corefiles
`large`	Create large corefiles (default)
`unlimited`	Create corefiles as large as necessary

Default Unlimited

Mode Global Configuration

Usage Core files are generated when a process crashes. The size of a core file can vary, its upper limit is controlled by this command. Files larger than this limit will be truncated by reducing the amount of stack and variable information stored.

Truncated core files may make debugging the failure difficult if not impossible. Reducing the amount of data stored in a core file is not recommended, however the facility is provided to reduce the amount of flash used.

Examples To restrict the size of the core file created, use the command:

```
        awplus# configure terminal
awplus(config)# exception coredump size small
```

To restore the size of the core files created to the default of `unlimited`, use the command:

```
        awplus# configure terminal
awplus(config)# no exception coredump size
```

log buffered

This command configures the device to store log messages in RAM. Messages stored in RAM are not retained on the device over a restart. Once the buffered log reaches its configured maximum allowable size old messages will be deleted to make way for new ones.

Syntax

```
log buffered

no log buffered
```

Default The buffered log is configured by default.

Mode Global Configuration

Examples To configured the device to store log messages in RAM use the following commands:

```
        awplus# configure terminal
awplus(config)# log buffered
```

To configure the device to not store log messages in a RAM buffer use the following commands:

```
        awplus# configure terminal
awplus(config)# no log buffered
```

Validation Commands show log config

Related Commands default log buffered
log buffered (filter)
log buffered size

log buffered (filter)

Use this command to create a filter to select messages to be sent to the buffered log. Selection can be based on the priority/ severity of the message, the program that generated the message, the logging facility used, a sub-string within the message or a combination of some or all of these.

The **no** variant of this command removes the corresponding filter, so that the specified messages are no longer sent to the buffered log.

Syntax

```
log buffered [level <level>] [program <program-name>]
    [facility <facility>] [msgtext <text-string>]
```

```
no log buffered [level <level>] [program <program-name>]
    [facility <facility>] [msgtext <text-string>]
```

Parameter	Description
`level`	Filter messages to the buffered log by severity level.
`<level>`	The minimum severity of message to send to the buffered log. The level can be specified as one of the following numbers or level names, where 0 is the highest severity and 7 is the lowest severity: 0 emergencies: System is unusable 1 alerts Action must be taken immediately 2 critical Critical conditions 3 errors Error conditions 4 warnings Warning conditions 5 notices Normal, but significant, conditions 6 informational Informational messages 7 debugging Debug-level messages
`program`	Filter messages to the buffered log by program. Include messages from a specified program in the buffered log.
`<program-name>`	The name of a program to log messages from, either one of the following predefined program names (not case-sensitive), or another program name (case-sensitive) that you find in the log output. rsvp Resource Reservation Protocol (RSVP) dot1x IEEE 802.1X Port-Based Access Control lacp Link Aggregation Control Protocol (LACP) stp Spanning Tree Protocol (STP) rstp Rapid Spanning Tree Protocol (RSTP) mstp Multiple Spanning Tree Protocol (MSTP) imi Integrated Management Interface (IMI) imish Integrated Management Interface Shell (IMISH) epsr Ethernet Protection Switched Rings (EPSR) irdp ICMP Router Discovery Protocol (IRDP) rmon Remote Monitoring loopprot Loop Protection dhcpsn DHCP snooping (DHCPSN)
`facility`	Filter messages to the buffered log by syslog facility.

Parameter	Description
<facility>	Specify one of the following syslog facilities to include messages from in the buffered log: kern Kernel messages user Random user-level messages mail Mail system daemon System daemons auth Security/authorization messages syslog Messages generated internally by syslogd lpr Line printer subsystem news Network news subsystem uucp UUCP subsystem cron Clock daemon authpriv Security/authorization messages (private) ftp FTP daemon
`msgtext`	Select messages containing a certain text string (maximum 128 characters).
<text-string>	A text string to match (maximum 128 characters). This is case sensitive, and must be the last text on the command line.

Default By default the buffered log has a filter to select messages whose severity level is "notices (5)" or higher. This filter may be removed using the **no** variant of this command.

Mode Global Configuration

Examples To add a filter to send all messages containing the text "`Bridging initialization`", to the buffered log use the following commands:

```
        awplus# configure terminal
awplus(config)# log buffered msgtext Bridging initialization
```

To remove a filter that sends all messages containing the text "`Bridging initialization`", to the buffered log use the following commands:

```
        awplus# configure terminal
awplus(config)# no log buffered msgtext Bridging initialization
```

Validation Commands show log config

Related Commands default log buffered
log buffered
log buffered size

log buffered size

This command configures the amount of memory that the buffered log is permitted to use. Once this memory allocation has been filled old messages will be deleted to make room for new messages.

Syntax

```
log buffered size <50-250>
```

Parameter	Description
<50-250>	Size of the RAM log in kilobytes

Mode Global Configuration

Example To allow the buffered log to use up to 100 kB of RAM use the following commands:

```
        awplus# configure terminal
awplus(config)# log buffered size 100
```

Validation Commands show log config

Related Commands default log buffered
log buffered

log console

This command configures the device to send log messages to consoles. The console log is configured by default to send messages to the devices main console port.

Use the **no** variant of this command to configure the device not to send log messages to consoles.

Syntax

```
log console

no log console
```

Mode Global Configuration

Examples To configure the device to send log messages use the following commands:

```
        awplus# configure terminal
awplus(config)# log console
```

To configure the device not to send log messages in all consoles use the following commands:

```
        awplus# configure terminal
awplus(config)# no log console
```

Validation Commands show log config

Related Commands log console (filter)

log console (filter)

This command creates a filter to select messages to be sent to all consoles when the log console command is given. Selection can be based on the priority/severity of the message, the program that generated the message, the logging facility used, a sub-string within the message or a combination of some or all of these.

Syntax

```
log console [level <level>] [program <program-name>]
    [facility <facility>] [msgtext <text-string>]

no log console [level <level>] [program <program-name>]
    [facility <facility>] [msgtext <text-string>]
```

Parameter	Description
`level`	Filter messages by severity level.
`<level>`	The minimum severity of message to send. The level can be specified as one of the following numbers or level names, where 0 is the highest severity and 7 is the lowest severity: 0 emergencies: System is unusable 1|alerts Action must be taken immediately 2|critical Critical conditions 3|errors Error conditions 4|warnings Warning conditions 5|notices Normal, but significant, conditions 6|informational Informational messages 7|debugging Debug-level messages
`program`	Filter messages by program. Include messages from a specified program.
`<program-name>`	The name of a program to log messages from, either one of the following predefined program names (not case-sensitive), or another program name (case-sensitive) that you find in the log output. rsvp Resource Reservation Protocol (RSVP) dot1x IEEE 802.1X Port-Based Access Control lacp Link Aggregation Control Protocol (LACP) stp Spanning Tree Protocol (STP) rstp Rapid Spanning Tree Protocol (RSTP) mstp Multiple Spanning Tree Protocol (MSTP) imi Integrated Management Interface (IMI) imish Integrated Management Interface Shell (IMISH) epsr Ethernet Protection Switched Rings (EPSR) irdp ICMP Router Discovery Protocol (IRDP) rmon Remote Monitoring loopprot Loop Protection dhcpsn DHCP snooping (DHCPSN)
`facility`	Filter messages to the buffered log by syslog facility.

Parameter	Description
<facility>	Specify one of the following syslog facilities to include messages from: kern Kernel messages user Random user-level messages mail Mail system daemon System daemons auth Security/authorization messages syslog Messages generated internally by syslogd lpr Line printer subsystem news Network news subsystem uucp UUCP subsystem cron Clock daemon authpriv Security/authorization messages (private) ftp FTP daemon
`msgtext`	Select messages containing a certain text string
<text-string>	A text string to match. This is case sensitive, and must be the last text on the command line.

Default By default the buffered log has a filter to select messages whose severity level is `critical` or higher. This filter may be removed using the **no** variant of this command. This filter may be removed and replaced by filters that are more selective.

Mode Global Configuration

Examples To create a filter to send all messages generated by MSTP that have a severity of `info` or higher to console instances where the log console command has been given, remove the default filter that includes everything use the following commands:

```
        awplus# configure terminal
awplus(config)# log console level info program mstp
```

and then use the command:

```
awplus(config)# log console level info program mstp
```

To create a filter to send all messages containing the text "`Bridging initialization`" to console instances where the log console command has been given use the following commands:

```
        awplus# configure terminal
awplus(config)# log console msgtext "Bridging initialization"
```

To remove a default filter that includes sending critical, alert and emergency level messages to the console use the following commands:

```
        awplus# configure terminal
awplus(config)# no log console level critical
```

Validation Commands show log config

Related Commands log console

log email

This command configures the device to send log messages to an email address. The email address is specified in this command.

Syntax
```
log email <email-address>
```

Parameter	Description
<email-address>	The email address to send log messages to

Default By default no filters are defined for email log targets. Filters must be defined before messages will be sent.

Mode Global Configuration

Example To have log messages emailed to the email address `admin@alliedtelesis.com` use the following commands:

```
        awplus# configure terminal
awplus(config)# log email admin@alliedtelesis.com
```

Validation Commands show log config

Related Commands default log email
log email

log email (filter)

This command creates a filter to select messages to be sent to an email address. Selection can be based on the priority/ severity of the message, the program that generated the message, the logging facility used, a sub-string within the message or a combination of some or all of these.

The **no** variant of this command configures the device to no longer send log messages to a specified email address. All configuration relating to this log target will be removed.

Syntax

```
log email <email-address> [level <level>] [program <program-name>]
    [facility <facility>] [msgtext <text-string>]
```

```
no log email <email-address> [level <level>] [program <program-name>]
    [facility <facility>] [msgtext <text-string>]
```

Parameter	Description
<email-address>	The email address to send logging messages to
level	Filter messages by severity level.
<level>	The minimum severity of messages to send. The level can be specified as one of the following numbers or level names, where 0 is the highest severity and 7 is the lowest severity: 0 emergencies: System is unusable 1 alerts Action must be taken immediately 2 critical Critical conditions 3 errors Error conditions 4 warnings Warning conditions 5 notices Normal, but significant, conditions 6 informational Informational messages 7 debugging Debug-level messages
program	Filter messages by program. Include messages from a specified program in the log.
<program-name>	The name of a program to log messages from, either one of the following predefined program names (not case-sensitive), or another program name (case -sensitive) that you find in the log output. rsvp Resource Reservation Protocol (RSVP) dot1x IEEE 802.1X Port-Based Access Control lacp Link Aggregation Control Protocol (LACP) stp Spanning Tree Protocol (STP) rstp Rapid Spanning Tree Protocol (RSTP) mstp Multiple Spanning Tree Protocol (MSTP) imi Integrated Management Interface (IMI) imish Integrated Management Interface Shell (IMISH) epsr Ethernet Protection Switched Rings (EPSR) irdp ICMP Router Discovery Protocol (IRDP) rmon Remote Monitoring loopprot Loop Protection dhcpsn DHCP snooping (DHCPSN)

Parameter	Description
`facility`	Filter messages to the log by syslog facility.
`<facility>`	Specify one of the following syslog facilities to include messages from in the log: kern Kernel messages user Random user-level messages mail Mail system daemon System daemons auth Security/authorization messages syslog Messages generated internally by syslogd lpr Line printer subsystem news Network news subsystem uucp UUCP subsystem cron Clock daemon authpriv Security/authorization messages (private) ftp FTP daemon
`msgtext`	Select messages containing a certain text string
`<text-string>`	A text string to match. This is case sensitive, and must be the last text on the command line.

Mode Global Configuration

Examples To create a filter to send all messages containing the text "`Bridging initialization`", to the email address `admin@homebase.com` use the following commands:

```
        awplus# configure terminal
awplus(config)# log email admin@homebase.com msgtext
                "Bridging initialization"
```

To create a filter to send messages with a severity level of `informational` and above to the email address `admin@alliedtelesis.com` use the following commands:

```
        awplus# configure terminal
awplus(config)# log email admin@alliedtelesis.com level
                informational
```

To stop the device emailing log messages emailed to the email address `admin@alliedtelesis.com` use the following commands:

```
        awplus# configure terminal
awplus(config)# no log email admin@homebase.com
```

To remove a filter that sends messages with a severity level of `informational` and above to the email address `admin@alliedtelesis.com` use the following commands:

```
        awplus# configure terminal

awplus(config)# no log email admin@alliedtelesis.com level
                informational
```

Related Commands

default log email
log email
show log config

log email time

This command configures the time used in messages sent to an email address. If the syslog server is in a different time zone to your switch then the time offset can be configured using either the **utc-offset** parameter option keyword or the **local-offset** parameter option keyword, where **utc-offset** is the time difference from UTC (Universal Time, Coordinated) and **local-offset** is the difference from local time.

Syntax

```
log email <email-address> time {local|local-offset|utc-offset
    {plus|minus}<0-24>}
```

Parameter	Description
`<email-address>`	The email address to send log messages to
`time`	Specify the time difference between the email recipient and the switch you are configuring.
`local`	The switch is in the same time zone as the email recipient
`local-offset`	The switch is in a different time zone to the email recipient. Use the **plus** or **minus** keywords and specify the difference (offset) from local time of the switch to the email recipient in hours.
`utc-offset`	The switch is in a different time zone to the email recipient. Use the **plus** or **minus** keywords and specify the difference (offset) from UTC time of the switch to the email recipient in hours.
`plus`	Negative offset (difference) from the switch to the email recipient.
`minus`	Positive offset (difference) from the switch to the email recipient.
`<0-24>`	World Time zone offset in hours

Default The default is **local** time.

Mode Global Configuration

Usage Use the **local** option if the email recipient is in the same time zone as this device. Messages will display the time as on the local device when the message was generated.

Use the **offset** option if the email recipient is in a different time zone to this device. Specify the time offset of the email recipient in hours. Messages will display the time they were generated on this device but converted to the time zone of the email recipient.

Examples To send messages to the email address `test@home.com` in the same time zone as the switch's local time zone, use the following commands:

```
        awplus# configure terminal
awplus(config)# log email admin@base.com time local 0
```

To send messages to the email address `admin@base.com` with the time information converted to the time zone of the email recipient, which is 3 hours ahead of the switch's local time zone, use the following commands:

```
        awplus# configure terminal
awplus(config)# log email admin@base.com time local-offset
                plus 3
```

To send messages to the email address `user@remote.com` with the time information converted to the time zone of the email recipient, which is 3 hours behind the switch's UTC time zone, use the following commands:

```
        awplus# configure terminal

awplus(config)# log email user@remote.com time utc-offset
                minus 3
```

Validation Commands show log config

Related Commands default log buffered

log host

This command configures the device to send log messages to a remote syslog server via UDP port 514. The IP address of the remote server must be specified. By default no filters are defined for remote syslog servers. Filters must be defined before messages will be sent.

Syntax

```
log host <ip-addr>

no log host <ip-addr>
```

Parameter	Description
<ip-addr>	The IP address of a remote syslog server in dotted decimal format A.B.C.D

Mode Global Configuration

Examples To configure the device to send log messages to a remote syslog server with IP address `10.32.16.99` use the following commands:

```
        awplus# configure terminal
awplus(config)# log host 10.32.16.99
```

To stop the device from sending log messages to the remote syslog server with IP address `10.32.16.99` use the following commands:

```
        awplus# configure terminal
awplus(config)# no log host 10.32.16.99
```

Validation Commands show log config

Related Commands default log host

log host (filter)

This command creates a filter to select messages to be sent to a remote syslog server. Selection can be based on the priority/severity of the message, the program that generated the message, the logging facility used, a substring within the message or a combination of some or all of these.

The **no** variant of this command configures the device to no longer send log messages to a remote syslog server. The IP address of the syslog server must be specified. All configuration relating to this log target will be removed.

Syntax

```
log host <ip-addr> [level <level>] [program <program-name>]
    [facility <facility>] [msgtext <text-string>]
```

```
no log host <ip-addr> [level <level>] [program <program-name>]
    [facility <facility>] [msgtext <text-string>]
```

Parameter	Description
`<ip-addr>`	The IP address of a remote syslog server
`level`	Filter messages by severity level.
`<level>`	The minimum severity of messages to send. The level can be specified as one of the following numbers or level names, where 0 is the highest severity and 7 is the lowest severity: 0|emergencies: System is unusable 1|alerts Action must be taken immediately 2|critical Critical conditions 3|errors Error conditions 4|warnings Warning conditions 5|notices Normal, but significant, conditions 6|informational Informational messages 7|debugging Debug-level messages
`program`	Filter messages by program. Include messages from a specified program in the log.
`<program-name>`	The name of a program to log messages from, either one of the following predefined program names (not case-sensitive), or another program name (case -sensitive) that you find in the log output. rsvp Resource Reservation Protocol (RSVP) dot1x IEEE 802.1X Port-Based Access Control

Parameter	Description
<program-name> (cont.)	lacp Link Aggregation Control Protocol (LACP) stp Spanning Tree Protocol (STP) rstp Rapid Spanning Tree Protocol (RSTP) mstp Multiple Spanning Tree Protocol (MSTP) imi Integrated Management Interface (IMI) imish Integrated Management Interface Shell (IMISH) epsr Ethernet Protection Switched Rings (EPSR) irdp ICMP Router Discovery Protocol (IRDP) rmon Remote Monitoring loopprot Loop Protection dhcpsn DHCP snooping (DHCPSN)
`facility`	Filter messages to the log by syslog facility.
<facility>	Specify one of the following syslog facilities to include messages from in the log: kern Kernel messages user Random user-level messages mail Mail system daemon System daemons auth Security/authorization messages syslog Messages generated internally by syslogd lpr Line printer subsystem news Network news subsystem uucp UUCP subsystem cron Clock daemon authpriv Security/authorization messages (private) ftp FTP daemon
`msgtext`	Select messages containing a certain text string
<text-string>	A text string to match. This is case sensitive, and must be the last text on the command line.

Mode Global Configuration

Examples To create a filter to send all messages containing the text "`Bridging initialization`", to a remote syslog server with IP address `10.32.16.21` use the following commands:

```
        awplus# configure terminal
awplus(config)# log host 10.32.16.21 msgtext "Bridging
                initialization"
```

To create a filter to send messages with a severity level of `informational` and above to the syslog server with IP address `10.32.16.21` use the following commands:

```
        awplus# configure terminal
awplus(config)# log host 10.32.16.21 level informational
```

To remove a filter that sends all messages containing the text "`Bridging initialization`", to a remote syslog server with IP address `10.32.16.21` use the following commands:

```
        awplus# configure terminal
awplus(config)# no log host 10.32.16.21 msgtext "Bridging
                initialization"
```

To remove a filter that sends messages with a severity level of `informational` and above to the syslog server with IP address `10.32.16.21` use the following commands:

```
 awplusawpluls# configure terminal
awplus(config)# no log host 10.32.16.21 level informational
```

Related Commands default log host
show log config

log host time

This command configures the time used in messages sent to a remote syslog server. If the syslog server is in a different time zone to your switch then the time offset can be configured using either the **utc-offset** parameter option keyword or the **local-offset** parameter option keyword, where **utc-offset** is the time difference from UTC (Universal Time, Coordinated) and **local-offset** is the difference from local time.

Syntax

```
log host <email-address> time {local|local-offset|utc-offset
   {plus|minus} <0-24>}
```

Parameter	Description
<email-address>	The email address to send log messages to
`time`	Specify the time difference between the email recipient and the switch you are configuring.
`local`	The switch is in the same time zone as the email recipient
`local-offset`	The switch is in a different time zone to the email recipient. Use the **plus** or **minus** keywords and specify the difference (offset) from local time of the switch to the email recipient in hours.
`utc-offset`	The switch is in a different time zone to the email recipient. Use the **plus** or **minus** keywords and specify the difference (offset) from UTC time of the switch to the email recipient in hours.
`plus`	Negative offset (difference) from the switch to the syslog server.
`minus`	Positive offset (difference) from the switch to the syslog server.
<0-24>	World Time zone offset in hours

Default The default is **local** time.

Mode Global Configuration

Usage Use the **local** option if the remote syslog server is in the same time zone as the switch. Messages will display the time as on the local device when the message was generated.

Use the **offset** option if the email recipient is in a different time zone to this device. Specify the time offset of the remote syslog server in hours. Messages will display the time they were generated on this device but converted to the time zone of the remote syslog server.

Examples To send messages to the remote syslog server with the IP address `10.32.16.21` in the same time zone as the switch's local time zone, use the following commands:

```
        awplus# configure terminal
awplus(config)# log host 10.32.16.21 time local 0
```

To send messages to the remote syslog server with the IP address `10.32.16.12` with the time information converted to the time zone of the remote syslog server, which is 3 hours ahead of the switch's local time zone, use the following commands:

```
        awplus# configure terminal
awplus(config)# log host 10.32.16.12 time local-offset plus 3
```

To send messages to the remote syslog server with the IP address `10.32.16.02` with the time information converted to the time zone of the email recipient, which is 3 hours behind the switch's UTC time zone, use the following commands:

```
        awplus# configure terminal
 awplus(config)# log host 10.32.16.02 time utc-offset minus 3
```

Validation Commands show log config

Related Commands default log buffered

log monitor (filter)

This command creates a filter to select messages to be sent to the terminal when the terminal monitor command is given. Selection can be based on the priority/severity of the message, the program that generated the message, the logging facility used, a sub-string within the message or a combination of some or all of these.

Syntax

```
log monitor [level <level>] [program <program-name>]
    [facility <facility>] [msgtext <text-string>]

no log monitor [level <level>] [program <program-name>]
    [facility <facility>] [msgtext <text-string>]
```

Parameter	Description
`level`	Filter messages to the permanent log by severity level.
`<level>`	The minimum severity of message to send to the log. The level can be specified as one of the following numbers or level names, where 0 is the highest severity and 7 is the lowest severity: 0|emergencies: System is unusable 1|alerts Action must be taken immediately 2|critical Critical conditions 3|errors Error conditions 4|warnings Warning conditions 5|notices Normal, but significant, conditions 6|informational Informational messages 7|debugging Debug-level messages
`program`	Filter messages to the permanent log by program. Include messages from a specified program in the log.
`<program-name>`	The name of a program to log messages from, either one of the following predefined program names (not case-sensitive), or another program name (case -sensitive) that you find in the log output. rsvp Resource Reservation Protocol (RSVP) dot1x IEEE 802.1X Port-Based Access Control lacp Link Aggregation Control Protocol (LACP) stp Spanning Tree Protocol (STP) rstp Rapid Spanning Tree Protocol (RSTP) mstp Multiple Spanning Tree Protocol (MSTP) imi Integrated Management Interface (IMI)
`<program-name>` `(cont.)`	imish Integrated Management Interface Shell (IMISH) epsr Ethernet Protection Switched Rings (EPSR) irdp ICMP Router Discovery Protocol (IRDP) rmon Remote Monitoring loopprot Loop Protection dhcpsn DHCP snooping (DHCPSN)
`facility`	Filter messages to the permanent log by syslog facility.

Parameter	Description
<*facility*>	Specify one of the following syslog facilities to include messages from in the log: kern Kernel messages user Random user-level messages mail Mail system daemon System daemons auth Security/authorization messages syslog Messages generated internally by syslogd lpr Line printer subsystem news Network news subsystem uucp UUCP subsystem cron Clock daemon authpriv Security/authorization messages (private) ftp FTP daemon
`msgtext`	Select messages containing a certain text string
<*text-string*>	A text string to match. This is case sensitive, and must be the last text on the command line.

Default By default there is a filter to select all messages. This filter may be removed and replaced by filters that are more selective.

Mode Global Configuration

Examples To create a filter to send all messages generated by MSTP that have a severity of `info` or higher to terminal instances where the terminal monitor command has been given use the following commands:

```
        awplus# configure terminal
awplus(config)# log monitor level info program mstp
```

To remove a default filter that includes sending everything to the terminal use the following commands:

```
        awplus# configure terminal
awplus(config)# no log monitor level debugging
```

Validation Commands show log config

Related Commands terminal monitor

log permanent

This command configures the device to send log messages to non-volatile storage (NVS) on the device. Log messages sent to NVS are retained on the device over a restart, that is they are permanent. Once the permanent log reaches its configured maximum allowable size old messages will be deleted to make way for new ones.

The **no** variant of this command configures the device not to send any messages to the permanent log. Log messages will not be retained over a restart.

Syntax

```
log permanent
```

```
no log permanent
```

Mode Global Configuration

Examples To enable permanent logging use the following commands:

```
         awplus# configure terminal
 awplus(config)# log permanent
```

To disable permanent logging use the following commands:

```
         awplus# configure terminal
 awplus(config)# no log permanent
```

Validation Commands show log config

Related Commands default log permanent
log permanent (filter)
log permanent size
show log permanent

log permanent (filter)

This command creates a filter to select messages to be sent to the permanent log. Selection can be based on the priority/ severity of the message, the program that generated the message, the logging facility used, a sub-string within the message or a combination of some or all of these.

The **no** variant of this command removes the corresponding filter, so that the specified messages are no longer sent to the permanent log.

Syntax

```
log permanent [level <level>] [program <program-name>]
    [facility <facility>] [msgtext <text-string>]

no log permanent [level <level>] [program <program-name>]
    [facility <facility>] [msgtext <text-string>]
```

Parameter	Description
`level`	Filter messages to the permanent log by severity level.
<level>	The minimum severity of message to send to the log. The level can be specified as one of the following numbers or level names, where 0 is the highest severity and 7 is the lowest severity: 0|emergencies: System is unusable 1|alerts Action must be taken immediately 2|critical Critical conditions 3|errors Error conditions 4|warnings Warning conditions 5|notices Normal, but significant, conditions 6|informational Informational messages 7|debugging Debug-level messages
`program`	Filter messages to the permanent log by program. Include messages from a specified program in the log.
<program-name>	The name of a program to log messages from, either one of the following predefined program names (not case-sensitive), or another program name (case -sensitive) that you find in the log output. rsvp Resource Reservation Protocol (RSVP) dot1x IEEE 802.1X Port-Based Access Control lacp Link Aggregation Control Protocol (LACP) stp Spanning Tree Protocol (STP) rstp Rapid Spanning Tree Protocol (RSTP) mstp Multiple Spanning Tree Protocol (MSTP) imi Integrated Management Interface (IMI) imish Integrated Management Interface Shell (IMISH) epsr Ethernet Protection Switched Rings (EPSR) irdp ICMP Router Discovery Protocol (IRDP) rmon Remote Monitoring loopprot Loop Protection dhcpsn DHCP snooping (DHCPSN)
`facility`	Filter messages to the permanent log by syslog facility.

Parameter	Description
<facility>	Specify one of the following syslog facilities to include messages from in the log: kern Kernel messages user Random user-level messages mail Mail system daemon System daemons auth Security/authorization messages syslog Messages generated internally by syslogd lpr Line printer subsystem news Network news subsystem uucp UUCP subsystem cron Clock daemon authpriv Security/authorization messages (private) ftp FTP daemon
`msgtext`	Select messages containing a certain text string
<text-string>	A text string to match. This is case sensitive, and must be the last text on the command line.

Default By default the buffered log has a filter to select messages whose severity level is `notices (5)` or higher. This filter may be removed using the **no** variant of this command.

Mode Global Configuration

Examples To create a filter to send all messages containing the text "`Bridging initialization`", to the permanent log use the following commands:

```
        awplus# configure terminal
awplus(config)# log permanent msgtext Bridging initialization
```

Validation Commands show log config

Related Commands default log permanent
log permanent
log permanent size
show log permanent

log permanent size

This command configures the amount of memory that the permanent log is permitted to use. Once this memory allocation has been filled old messages will be deleted to make room for new messages.

Syntax

```
log permanent size <50-250>
```

Parameter	Description
<50-250>	Size of the permanent log in kilobytes

Mode Global Configuration

Example To allow the permanent log to use up to 100 kB of NVS use the following commands:

```
        awplus# configure terminal
awplus(config)# log permanent size 100
```

Validation Commands show log config

Related Commands default log permanent
log permanent

log-rate-limit nsm

This command limits the number of log messages generated by the switch for a given interval.

Use the **no** variant of this command to revert to the default number of log messages generated by the switch of up to 200 log messages per second.

Syntax

```
log-rate-limit nsm messages <message-limit> interval <time-interval>

no log-rate-limit nsm
```

Parameter	Description
<message-limit>	<1-65535> The number of log messages generated by the switch.
<time-interval>	<0-65535> The time period for log message generation in 1/100 seconds. If an interval of 0 is specified then no log message rate limiting is applied.

Default By default, the switch will allow 200 log messages to be generated per second.

Mode Global Configuration

Usage Previously, if the switch received a continuous stream of IGMP packets with errors, such as when a packet storm occurs because of a network loop, then the switch generates a lot of log messages using more and more memory, which may ultimately cause the switch to shutdown. This log rate limiting feature constrains the rate that log messages are generated by the switch.

Note that if within the given time interval, the number of log messages exceeds the limit, then any excess log messages are discarded. At the end of the time interval, a single log message is generated indicating that log messages were discarded due to the log rate limit being exceeded.

Thus if the expectation is that there will be a lot of discarded log messages due to log rate limiting, then it is advisable to set the time interval to no less than 100, which means that there would only be one log message, indicating log excessive log messages have been discarded.

Examples To limit the switch to generate up to 300 log messages per second, use the following commands:

```
        awplus# configure terminal
awplus(config)# log-rate-limit nsm messages 300 interval 100
```

To return the switch the default setting, to generate up to 200 log messages per second, use the following commands:

```
        awplus# configure terminal
awplus(config)# no log-rate-limit nsm
```

show counter log

This command displays log counter information.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax `show counter log`

Mode User Exec and Privileged Exec

Example To display the log counter information, use the command:

```
awplus# show counter log
```

Output Figure 10-1: Example output from the **show counter log** command

```
Log counters
Total Received         ......... 2328
Total Received P0      ......... 0
Total Received P1      ......... 0
Total Received P2      ......... 1
Total Received P3      ......... 9
Total Received P4      ......... 32
Total Received P5      ......... 312
Total Received P6      ......... 1602
Total Received P7      ......... 372
```

Table 10-1: Parameters in output of the **show counter log** command

Parameter	Description
`Total Received`	Total number of messages received by the log
`Total Received P0`	Total number of Priority 0 (Emergency) messages received
`Total Received P1`	Total number of Priority 1 (Alert) messages received
`Total Received P2`	Total number of Priority 2 (Critical) messages received
`Total Received P3`	Total number of Priority 3 (Error) messages received
`Total Received P4`	Total number of Priority 4 (Warning) messages received
`Total Received P5`	Total number of Priority 5 (Notice) messages received
`Total Received P6`	Total number of Priority 6 (Info) messages received
`Total Received P7`	Total number of Priority 7 (Debug) messages received

Related Commands show log config

show exception log

This command displays the contents of the exception log. When used within a virtual chassis stack (VCS), this command will display the contents of the exception log for all the stack members.

Syntax `show exception log`

Mode User Exec and Privileged Exec

Example To display the exception log, use the command:

```
awplus# show exception log
```

Output Figure 10-2: Example output from the **show exception log** command

```
awplus#show exception log

Stack member 1:

<date> <time> <facility>.<severity> <program[<pid>]>: <message>
-------------------------------------------------------------------------
2012 May 29 04:08:46 local7.debug awplus corehandler: Process imish (PID:2200) s
ignal 5, core dumped to /flash/imish-r1-main-xinz-1212034124-2200.tgz
2012 May 29 04:10:21 local7.debug awplus corehandler: Process stackd (PID:1136)
signal 5, core dumped to /flash/stackd-r1-main-xinz-1212034216-1136.tgz
-------------------------------------------------------------------------
Stack member 2:

<date> <time> <facility>.<severity> <program[<pid>]>: <message>
-------------------------------------------------------------------------
2012 Mar 28 03:15:32 local7.debug awplus corehandler: Process imish (PID:2253) s
ignal 5, core dumped to /flash/imish-r1-main-xinz-7442130-2253.tgz
2012 Mar 28 03:16:06 local7.debug awplus corehandler: Process imish (PID:2416) s
ignal 5, core dumped to /flash/imish-r1-main-xinz-7442165-2416.tgz
2012 Mar 28 03:17:33 local7.debug awplus corehandler: Process aisexec (PID:1786)
 signal 5, core dumped to /flash/aisexec-r1-main-xinz-7442251-1786.tgz
-------------------------------------------------------------------------
```

show log

This command displays the contents of the buffered log.

For information on output options, see “Controlling “show” Command Output” on page 1.41.

Syntax

```
show log [tail [<10-250>]]
```

Parameter	Description
`tail`	Display only the latest log entries.
`<10-250>`	Specify the number of log entries to display.

Default By default the entire contents of the buffered log is displayed.

Mode User Exec, Privileged Exec and Global Configuration

Usage If the optional **tail** parameter is specified only the latest 10 messages in the buffered log are displayed. A numerical value can be specified after the **tail** parameter to select how many of the latest messages should be displayed.

Examples To display the contents of the buffered log use the command:

```
awplus# show log
```

To display the 10 latest entries in the buffered log use the command:

```
awplus# show log tail 10
```

Output Figure 10-3: Example output from the **show log** command

```
awplus#show log

<date> <time> <facility>.<severity> <program[<pid>]>: <message>
-------------------------------------------------------------------
2011 Aug 29 07:55:22 kern.notice awplus kernel: Linux version 2.6.32.12-at1 (mak
er@awpmaker03-dl) (gcc version 4.3.3 (Gentoo 4.3.3-r3 p1.2, pie-10.1.5) ) #1 Wed
 Dec 8 11:53:40 NZDT 2010
2011 Aug 29 07:55:22 kern.warning awplus kernel: No pci config register base in
dev tree, using default
2011 Aug 29 07:55:23 kern.notice awplus kernel: Kernel command line: console=tty
S0,9600 releasefile=x510-5.4.2A.rel ramdisk=14688 bootversion=1.1.0-rc12
loglevel=1
 extraflash=00000000
2011 Aug 29 07:55:25 kern.notice awplus kernel: RAMDISK: squashfs filesystem fou
nd at block 0
2011 Aug 29 07:55:28 kern.warning awplus kernel: ipifwd: module license 'Proprie
tary' taints kernel.
.
.
.
```

Figure 10-4: Example output from the **show log tail** command

```
awplus#show log tail

<date> <time> <facility>.<severity> <program[<pid>]>: <message>
--------------------------------------------------------------------
2006 Nov 10 13:30:01 cron.notice crond[116]: USER manager pid 469 cmd logrotate /
etc/logrotate.conf
2006 Nov 10 13:30:01 cron.notice crond[116]: USER manager pid 471 cmd nbqueue --
wipe
2006 Nov 10 13:35:01 cron.notice crond[116]: USER manager pid 472 cmd nbqueue --
wipe
2006 Nov 10 13:40:01 cron.notice crond[116]: USER manager pid 477 cmd nbqueue --
wipe
2006 Nov 10 13:44:36 syslog.notice syslog-ng[67]: Log statistics;
processed=\'center(queued)=70\', processed=\'2006 Nov 10 13:45:01 cron.notice
crond[116]: USER manager pid 478 cmd logrotate /etc/logrotate.conf
2006 Nov 10 13:45:01 cron.notice crond[116]: USER manager pid 480 cmd nbqueue --
wipe
2006 Nov 10 13:49:32 syslog.notice syslog-ng[67]: SIGHUP received, reloading
configuration;
2006 Nov 10 13:50:01 cron.notice crond[116]: USER manager pid 482 cmd nbqueue --
wipe
2006 Nov 10 13:55:01 cron.notice crond[116]: USER manager pid 483 cmd nbqueue --
wipe
.
.
.
```

Related Commands show log config
show log permanent

show log config

This command displays information about the logging system. This includes the configuration of the various log destinations, buffered, permanent, syslog servers (hosts) and email addresses. This also displays the latest status information for each of these destinations.

Syntax

```
show log config
```

Mode User Exec, Privileged Exec and Global Configuration

Example To display the logging configuration use the command:

```
awplus# show log config
```

Output Figure 10-5: Example output from the **show log config** command

```
Buffered log:
Status ......... enabled
  Maximum size ... 100kb
  Filters:
 *1 Level ........ notices
    Program ...... any
    Facility ..... any
    Message text . any
  2 Level ........ informational
    Program ...... mstp
    Facility ..... daemon
    Message text . any
  Statistics ..... 1327 messages received, 821 accepted by filter (2006 Dec 11
10:36:16)
Permanent log:
  Status ......... enabled
  Maximum size ... 60kb
  Filters:
  1 Level ........ error
    Program ...... any
    Facility ..... any
    Message text . any
 *2 Level ........ warnings
    Program ...... dhcp
    Facility ..... any
    Message text . "pool exhausted"
  Statistics ..... 1327 messages received, 12 accepted by filter (2006 Dec 11
10:36:16)
Host 10.32.16.21:
  Time offset .... +2:00
  Offset type .... UTC
  Filters:
  1 Level ........ critical
    Program ...... any
    Facility ..... any
    Message text . any
  Statistics ..... 1327 messages received, 1 accepted by filter (2006 Dec 11
10:36:16)
Email admin@alliedtelesis.com:
  Time offset .... +0:00
  Offset type .... Local
  Filters:
  1 Level ........ emergencies
    Program ...... any
    Facility ..... any
    Message text . any
  Statistics ..... 1327 messages received, 0 accepted by filter (2006 Dec 11
10:36:16)
Monitor log:
  Filters:
 *1   Level ...... debugging
      Program .... any
      Facility ... any
      Msg text ... any
  Statistics ..... Not available
Console log:
  Status ......... enabled
  List of consoles:
  1 .............. ttyS0
  Filters:
 *1   Level ...... critical
      Program .... any
      Facility ... any
      Msg text ... any
  Statistics ..... 1327 messages received, 1 accepted by filter (2006 Dec 11
10:36:16)
```

In the above example the '*' next to filter 1 in the buffered log configuration indicates that this is the default filter. The permanent log has had its default filter removed, so none of the filters are marked with ''*'.

Note Statistics are updated periodically not in real time. Whenever logging configuration commands are issued the statistics are reset. Whenever automatic log rotation occurs the statistics are reset

Note Terminal log and console log cannot be set at the same time. If console logging is enabled then the terminal logging is turned off.

Related Commands

show counter log
show log
show log permanent

show log permanent

This command displays the contents of the permanent log.

When used within a virtual chassis stack (VCS), this command will display the contents of the permanent log for all the stack members.

Syntax

```
show log permanent [tail [<10-250>]]
```

Parameter	Description
`tail`	Display only the latest log entries
`<10-250>`	Specify the number of log entries to display

Default If the optional **tail** parameter is specified only the latest 10 messages in the permanent log are displayed. A numerical value can be specified after the **tail** parameter to select how many of the latest messages should be displayed.

Mode User Exec, Privileged Exec and Global Configuration

Example To display the permanent log, use the command:

```
awplus# show log permanent
```

To display the 10 latest entries in the permanent log, use the command:

```
awplus# show log permanent tail
```

Output Figure 10-6: Example output from the **show log permanent** command

```
<date> <time> <facility>.<severity> <program[<pid>]>: <message>
-----------------------------------------------------------------
2006 Nov 10 09:30:09 syslog.notice syslog-ng[67]: syslog-ng starting up;
version=\'2.0rc3\'
2006 Nov 10 09:30:09 auth.warning portmap[106]: user rpc not found, reverting to
user bin
2006 Nov 10 09:30:09 cron.notice crond[116]: crond 2.3.2 dillon, started, log
level 8
2006 Nov 10 09:30:14 daemon.err snmpd[181]: /flash/.configs/snmpd.conf: line 20:
Error: bad SUBTREE object 2006 Nov 10 09:30:14 user.info HSL[192]: HSL: INFO:
Registering port port1.0.1
.
.
.
```

Figure 10-7: Example output from the **show log permanent tail** command

```
awplus>show log permanent tail

<date> <time> <facility>.<severity> <program[<pid>]>: <message>
------------------------------------------------------------------------
2006 Nov 10 13:30:01 cron.notice crond[116]: USER manager pid 469 cmd logrotate /
etc/logrotate.conf
2006 Nov 10 13:30:01 cron.notice crond[116]: USER manager pid 471 cmd nbqueue --
wipe
2006 Nov 10 13:35:01 cron.notice crond[116]: USER manager pid 472 cmd nbqueue --
wipe
2006 Nov 10 13:40:01 cron.notice crond[116]: USER manager pid 477 cmd nbqueue --
wipe
2006 Nov 10 13:44:36 syslog.notice syslog-ng[67]: Log statistics;
processed=\'center(queued)=70\', processed=\'2006 Nov 10 13:45:01 cron.notice
crond[116]: USER manager pid 478 cmd logrotate /etc/logrotate.conf
2006 Nov 10 13:45:01 cron.notice crond[116]: USER manager pid 480 cmd nbqueue --
wipe
2006 Nov 10 13:49:32 syslog.notice syslog-ng[67]: SIGHUP received, reloading
configuration;
2006 Nov 10 13:50:01 cron.notice crond[116]: USER manager pid 482 cmd nbqueue --
wipe
2006 Nov 10 13:55:01 cron.notice crond[116]: USER manager pid 483 cmd nbqueue --
wipe
------------------------------------------------------------------------
.
.
.
```

Figure 10-8: Example output from the **show log permanent** command for a stack

```
Stack member 1:

<date> <time> <facility>.<severity> <program[<pid>]>: <message>
-------------------------------------------------------------------------
2008 May 28 23:11:21 user.crit awplus-2 VCS[1190]: Member 2 (00-00-cd-24-ff-57)
has become the Active Master
2008 May 28 23:11:21 daemon.warning awplus-2 rpc.statd[1300]: gethostbyname erro
r for awplus-2
2008 May 28 23:11:21 daemon.warning awplus-2 rpc.statd[1304]: gethostbyname erro
r for awplus-2
2008 May 28 23:11:53 user.err awplus NSM[1950]: VRRP Error: Can't set pktinfo
2008 May 29 03:54:40 user.alert awplus corerotate[26733]: Exception information
saved to flash:/imish-r1-main-xinz-7441248-19868.tgz
2008 May 29 03:55:47 user.crit awplus-1 VCS[1143]: Contact with the Active Maste
r has been lost
2008 May 29 03:55:47 user.crit awplus-1 VCS[1143]: Member 1 (00-09-41-fb-c3-0f)
has become the Disabled Master
2008 May 29 03:55:47 daemon.err awplus-1 mountd[1282]: Caught signal 15, un-regi
stering and exiting.
2008 May 29 03:55:47 user.warning awplus-1 kernel: nfsd: last server has exited
2008 May 29 03:55:47 user.warning awplus-1 kernel: nfsd: unexporting all filesys
tems



Stack member 2:

<date> <time> <facility>.<severity> <program[<pid>]>: <message>
-------------------------------------------------------------------------
2008 Mar 27 22:17:33 user.crit awplus-1 VCS[1143]: Member 2 (00-00-cd-24-ff-57)
has become the Active Master
2008 Mar 27 22:17:38 daemon.warning awplus-1 rpc.statd[1271]: gethostbyname erro
r for awplus-1
2008 Mar 27 22:17:38 daemon.warning awplus-1 rpc.statd[1276]: gethostbyname erro
r for awplus-1
2008 Mar 28 03:00:52 user.alert awplus corerotate[26733]: Exception information
saved to flash:/imish-r1-main-xinz-7441248-19868.tgz
2008 Mar 28 03:01:59 user.warning awplus NSM[1950]: imi_client_send_xem_removal
port1.0.1
2008 Mar 28 03:01:59 user.warning awplus NSM[1950]: imi_client_send_xem_removal
port1.2.1
2008 Mar 28 03:15:34 user.alert awplus corerotate[2380]: Exception information s
aved to flash:/imish-r1-main-xinz-7442130-2253.tgz
2008 Mar 28 03:16:08 user.alert awplus corerotate[2495]: Exception information s
aved to flash:/imish-r1-main-xinz-7442165-2416.tgz
2008 Mar 28 03:17:31 user.err awplus VCS[1200]: HA event handling failed with re
sult 9
2008 Mar 28 03:17:31 user.err awplus HSL[1246]: connection to aisexec lost (9)
2008 Mar 28 03:17:31 user.err awplus 802.1X[1451]: connection to aisexec lost (9
```

Related Commands remote-command <1-4> show
show log

show running-config log

This command displays the current running configuration of the Log utility.

Syntax

```
show running-config log
```

Mode Privileged Exec

Example To display the current configuration of the log utility, use the command:

```
awplus# show running-config log
```

Related Commands show log
show log config

Chapter 11: Scripting Commands

Command List

This chapter provides commands used for command scripts.

activate

This command activates a script file.

Syntax

```
activate [background] <script>
```

Parameter	Description
`background`	Activate a script to run in the background. A process that is running in the background will operate as a separate task, and will not interrupt foreground processing. Generally, we recommend running short, interactive scripts in the foreground and longer scripts in the background. The default is to run the script in the foreground.
`<script>`	The file name of the script to activate. The script is a command script consisting of commands documented in this software reference. Note that you must use either a **.scp** or a **.sh** filename extension for a valid script text file, as described below in the usage section for this command.

Mode Privileged Exec

Usage In a stacked environment you can use the CLI on a stack master to access file systems that are located on a member device. In this case the command specifies a file on the slave device. The slave's file system will be denoted by: `<hostname>-<member-id>` For example, **awplus-1** for member 1, **awplus-2** for member 2 etc.

When a script is activated, the privilege level is set to 1 enabling User Exec commands to run in the script. If you need to run Privileged Exec commands in your script you need to add an enable (Privileged Exec mode) command to the start of your script. If you need to run Global Configuration commands in your script you need to add a configure terminal command after the **enable** command at the start of your script.

The **activate** command executes the script in a new shell. A terminal length shell command, such as **terminal length 0** may also be required to disable a delay that would pause the display.

A script must be a text file with a filename extension of either **.sh** or **.scp** only for the AlliedWare Plus™ CLI to activate the script file. The **.sh** filename extension indicates the file is an ASH script, and the **.scp** filename extension indicates the file is an AlliedWare Plus™ script.

Examples To activate a command script to run as a background process, use the command:

```
awplus# activate background test.scp
```

To activate a script `/flash:/test.scp` in stack member 2, use the command:

```
awplus-2# activate awplus-2/flash:/test.scp
```

Related Commands configure terminal
echo
enable (Privileged Exec mode)
wait

echo

This command echoes a string to the terminal, followed by a blank line.

Syntax `echo <line>`

Parameter	Description
`<line>`	The string to echo

Mode User Exec and Privileged Exec

Usage This command may be useful in CLI scripts, to make the script print user-visible comments.

Example To echo the string `Hello World` to the console, use the command:

```
awplus# echo Hello World
```

```
Hello World
```

Related Commands activate
wait

wait

This command pauses execution of the active script for the specified period of time.

Syntax `wait <delay>`

Parameter	Description
<delay>	<1-65335> Specify the time delay in seconds

Default No wait delay is specified by default to pause script execution.

Mode Privileged Exec (when executed from a script not directly from the command line)

Usage Use this command to pause script execution in an **.scp** (AlliedWare Plus™ script) or an **.sh** (ASH script) file executed by the activate command. The script must contain an enable (Privileged Exec mode) command since the **wait** command is only executed in the Privileged Exec mode.When a script is activated, the privilege level is set to 1 enabling User Exec commands to run in the script. If you need to run Privileged Exec commands in your script you need to add an enable (Privileged Exec mode) command to the start of your script.

Example See an example **.scp** script file extract below that will show port counters for interface `port1.0.1` over a 10 second interval:

```
enable
show interface port1.0.1
wait 10
show interface port1.0.1
```

Related Commands activate
echo
enable (Privileged Exec mode)

Chapter 12: Interface Commands

Command List

This chapter provides an alphabetical reference of commands used to configure and display interfaces.

description (interface)

Use this command to add a description to a specific port or interface.

Syntax

```
description <description>
```

Parameter	Description
<description>	Text describing the specific interface.

Mode Interface Configuration

Example The following example uses this command to describe the device that a switch port is connected to.

```
         awplus# configure terminal
  awplus(config)# interface port1.0.2
awplus(config-if)# description Boardroom PC
```

interface (to configure)

Use this command to select one or more interfaces to configure.

Syntax

```
interface <interface-list>

interface lo
```

Parameter	Description
<*interface-list*>	The interfaces or ports to configure. An interface-list can be: ■ an interface (e.g. `vlan2`), a switch port (e.g. `port1.0.12`), a static channel group (e.g. `sa3`) or a dynamic (LACP) channel group (e.g. `po4`) ■ a continuous range of interfaces, ports, static channel groups or dynamic (LACP) channel groups separated by a hyphen; e.g. `vlan2-8`, or `port1.0.1-1.0.24`, or `sa2-4`, or `po1-3` ■ a comma-separated list of the above; e.g. `port1.0.1,port1.0.8-1.0.24`. Do not mix interface types in a list The specified interfaces must exist.
`lo`	The local loopback interface.

Usage A local loopback interface is one that is always available for higher layer protocols to use and advertise to the network. Although a local loopback interface is assigned an IP address, it does not have the usual requirement of connecting to a lower layer physical entity. This lack of physical attachment creates the perception of a local loopback interface always being accessible via the network.

Local loopback interfaces can be utilized by a number of protocols for various purposes. They can be used to improve access to the switch and also increase its reliability, security, scalability and protection. In addition, local loopback interfaces can add flexibility and simplify management, information gathering and filtering.

Mode Global Configuration

Example The following example shows how to enter Interface mode to configure `vlan1`. Note how the prompt changes.

```
          awplus# configure terminal
  awplus(config)# interface vlan1
awplus(config-if)#
```

The following example shows how to enter Interface mode to configure the local loopback interface.

```
          awplus# configure terminal
  awplus(config)# interface lo
awplus(config-if)#
```

Related Commands ip address
show interface
show interface brief

mru

Use this command to set the Maximum Receive Unit (MRU) size for switch ports, where MRU is the maximum frame size that switch ports can receive.

Use the **no** variant of this command to remove a previously specified Maximum Receive Unit (MRU) size for switch ports, and restore the default MRU size (1500 bytes) for switch ports.

Syntax

```
mru <mru-size>
```

```
no mru
```

Parameter	Description
<mru-size>	<68-16357> Specifies the Maximum Receive Unit (MRU) size in bytes, where: ■ 1500 bytes is the default Ethernet MRU size for an interface.

Default The default MRU size is 1500 bytes for switch ports.

Mode Interface Configuration for switch ports.

Usage Note that show interface output will only show MRU size for switch ports.

Examples To configure an MRU of 16357 bytes on `port1.0.2`, use the commands:

```
         awplus# configure terminal
 awplus(config)# interface port1.0.2
awplus(config-if)# mru 16357
```

To configure an MRU of 1500 bytes on `port1.0.2` to `port1.0.4` use the commands:

```
         awplus# configure terminal
 awplus(config)# interface port1.0.2-port1.0.4
awplus(config-if)# mru 1500
```

To restore the MRU size of 1500 bytes on `port1.0.2`, use the commands:

```
         awplus# configure terminal
 awplus(config)# interface port1.0.2
awplus(config-if)# no mru
```

Related Commands show interface

mtu

Use this command to set the Maximum Transmission Unit (MTU) size for VLANs, where MTU is the maximum packet size that VLANs can transmit. The MTU size setting is applied to both IPv4 and IPv6 packet transmission.

Use the **no** variant of this command to remove a previously specified Maximum Transmission Unit (MTU) size for VLANs, and restore the default MTU size (1500 bytes) for VLANs.

Syntax

```
mtu <mtu-size>
```

```
no mtu
```

Parameter	Description
<mtu-size>	<68-1500> Specifies the Maximum Transmission Unit (MTU) size in bytes, where 1500 bytes is the default Ethernet MTU size for an interface.

Default The default MTU size is 1500 bytes for VLAN interfaces.

Mode Interface Configuration for VLAN interfaces.

Usage If a switch receives an IPv4 packet for Layer 3 switching to another VLAN with an MTU size smaller than the packet size, and if the packet has the '**don't fragment**' bit set, then the switch will send an ICMP '**destination unreachable**' (3) packet type and a '**fragmentation needed and DF set**' (4) code back to the source. For IPv6 packets bigger than the MTU size of the transmitting VLAN interface, an ICMP '**packet too big**' (ICMP type 2 code 0) message is sent to the source.

MTU size can only be set for VLANs whose member ports are all non-trunked ports. If a trunked port moves to another VLAN then the trunked port's MTU size will not be set to the VLAN's MTU size, but will instead be set to the default MTU size of 1500 bytes.

Note that show interface output will only show MTU size for VLAN interfaces.

Examples To configure an MTU size of 1500 bytes on interface `vlan2`, use the commands:

```
        awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# mtu 1500
```

To configure an MTU size of 1500 bytes on interfaces `vlan2 to vlan4`, use the commands:

```
        awplus# configure terminal
awplus(config)# interface vlan2-vlan4
awplus(config-if)# mtu 1500
```

To restore the MTU size to the default MTU size of 1500 bytes on `vlan2`, use the commands

```
          awplus# configure terminal
  awplus(config)# interface vlan2
awplus(config-if)# no mtu
```

To restore the MTU size to the default MTU size of 1500 bytes on `vlan2` and `vlan4`, use the commands

```
          awplus# configure terminal
  awplus(config)# interface vlan2-vlan4
awplus(config-if)# no mtu
```

Related Commands show interface

show interface

Use this command to display interface configuration and status.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax

```
show interface [<interface-list>]

show interface lo
```

Parameter	Description
<interface-list>	The interfaces or ports to configure. An interface-list can be: ■ an interface (e.g. `vlan2`), a switch port (e.g. `port1.0.12`), a static channel group (e.g. `sa3`) or a dynamic (LACP) channel group (e.g. `po4`) ■ a continuous range of interfaces, ports, static channel groups or dynamic (LACP) channel groups separated by a hyphen; e.g. `vlan2-8`, or `port1.0.1-1.0.24`, or `sa2-4`, or `po1-3` ■ a comma-separated list of the above; e.g. `port1.0.1,port1.0.8-1.0.24`. Do not mix interface types in a list The specified interfaces must exist.
`lo`	The local loopback interface.

Mode User Exec and Privileged Exec

Usage Note that the output displayed with this command will show MTU (Maximum Transmission Unit) size for VLAN interfaces, and MRU (Maximum Received Unit) size for switch ports.

Examples To display configuration and status information for interfaces `port1.0.1` and `port1.0.4`, use the command:

```
awplus# show interface port1.0.1,port1.0.4
```

Figure 12-1: Example output from the **show interface** command

```
Interface port1.0.1
  Scope: both
  Link is UP, administrative state is UP
  Thrash-limiting
    Status Not Detected, Action learn-disable, Timeout 1(s)
  Hardware is Ethernet, address is 0000.f427.75a1
  index 6001 metric 1 mru 1500
  current duplex full, current speed 1000, current polarity mdix
  configured duplex auto, configured speed auto, configured polarity auto
  <UP,BROADCAST,RUNNING,MULTICAST>
  SNMP link-status traps: Disabled
    input packets 2, bytes 128, dropped 0, multicast packets 2
    output packets 387, bytes 24768, multicast packets 387 broadcast packets 0
  Time since last state change: 0 days 00:12:43
```

To display configuration and status information for interface `lo`, use the command:

```
awplus# show interface lo
```

Figure 12-2: Example output from the **show interface lo** command

```
Interface lo
  Scope: both
  Link is UP, administrative state is UP
  Hardware is Loopback
  index 1 metric 1
  <UP,LOOPBACK,RUNNING>
  VRF Binding: Not bound
  SNMP link-status traps: Disabled
    input packets 0, bytes 0, dropped 0, multicast packets 0
    output packets 0, bytes 0, multicast packets 0 broadcast packets 0
  Time since last state change: 69 days 01:28:47
```

To display configuration and status information for interfaces `vlan1` and `vlan2`, use the command:

```
awplus# show interface vlan1,vlan2
```

Figure 12-3: Example output from the **show interface vlan1,vlan2** command

```
Interface vlan1
  Scope: both
  Link is UP, administrative state is UP
  Hardware is VLAN, address is 0015.77e9.5c50
  IPv4 address 192.168.1.1/24 broadcast 192.168.1.255
  index 201 metric 1 mtu 1500
  arp ageing timeout 300
  <UP,BROADCAST,RUNNING,MULTICAST>
  VRF Binding: Not bound
  SNMP link-status traps: Disabled
  Bandwidth 1g
    input packets 295606, bytes 56993106, dropped 5, multicast packets 156
    output packets 299172, bytes 67379392, multicast packets 0 broadcast packets 0
  Time since last state change: 0 days 14:22:39
Interface vlan2
  Scope: both
  Link is DOWN, administrative state is UP
  Hardware is VLAN, address is 0015.77e9.5c50
  IPv4 address 192.168.2.1/24 broadcast 192.168.2.255
  Description: ip_phone_vlan
  index 202 metric 1 mtu 1500
  arp ageing timeout 300
  <UP,BROADCAST,MULTICAST>
  VRF Binding: Not bound
  SNMP link-status traps: Disabled
  Bandwidth 1g
    input packets 0, bytes 0, dropped 0, multicast packets 0
    output packets 90, bytes 4244, multicast packets 0 broadcast packets 0
  Time since last state change: 0 days 14:22:39
```

Related Commands mtu
show interface brief

show interface brief

Use this command to display brief interface, configuration, and status information, including provisioning information.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax `show interface brief`

Mode User Exec and Privileged Exec

Output Figure 12-4: Example output from the **show interface brief** command

```
awplus#show int brief
Interface              Status          Protocol
port1.0.1              admin up        down
port1.0.2              admin up        down
port1.0.3              admin up        down
port1.0.4              admin up        down
.
.
port1.0.23             admin up        down
port1.0.24             admin up        running
lo                     admin up        running
vlan1                  admin up        down
vlan2                  admin up        down
```

Table 12-1: Parameters in the output of the **show interface brief** command

Parameter	Description
`Interface`	The name or type of interface.
`Status`	The administrative state. This can be either **admin up** or **admin down**
`Protocol`	The link state. This can be either **down**, **running**, or **provisioned**

Related Commands show interface
show interface memory

show interface status

Use this command to display the status of the specified interface or interfaces. Note that when no interface or interfaces are specified then the status of all interfaces on the switch are shown.

Syntax `show interface [<port-list>] status`

Parameter	Description
<port-list>	The ports to display information about. The port list can be: ■ a switch port (e.g. `port1.0.12`) a static channel group (e.g. `sa3`) or a dynamic (LACP) channel group (e.g. `po3`) ■ a continuous range of ports separated by a hyphen, e.g. `port1.0.1-1.0.24`, or `sa1-2`, or `po1-4` ■ a comma-separated list of ports and port ranges, e.g. `port1.0.1,port1.0.4-1.2.24`. Do not mix switch ports, static channel groups, and dynamic (LACP) channel groups in the same list

Examples

Figure 12-5: Example output from the **show interface <port-list> status** command

```
awplus#show interface port1.0.1 -1.0.5 status
Port       Name               Status            Vlan Duplex   Speed Type
port1.0.1                     notconnect           1 auto      auto 1000BASE-T
port1.0.2                     notconnect           1 auto      auto 1000BASE-T
port1.0.3                     notconnect           1 auto      auto 1000BASE-T
port1.0.4                     notconnect           1 auto      auto 1000BASE-T
port1.0.5                     notconnect           1 auto      auto 1000BASE-T
```

To display the status of all ports, use the commands:

```
awplus# show interface status
```

Figure 12-6: Example output from the **show interface status** command

```
awplus#sho int status
Port      Name               Status             Vlan Duplex   Speed Type
port1.0.1 Trunk_Net          connected         trunk a-full  a-1000 1000BaseTX
port1.0.2 Access_Net1        connected             5 full       100 1000BaseTX
port1.0.3 Access_Net1        disabled              5 auto      auto 1000BaseTX
port1.0.4 Access_Net2        connected             6 a-half   a-100 1000BaseTX
port1.0.5 Private_Prom       connected            10 a-full   a-100 1000BaseTX
port1.0.6 Private_Net1       connected         10,11 a-full   a-100 1000BaseTX
port1.0.7 Private_Net2       connected         10,12 a-full   a-100 1000BaseTX
port1.0.8                    notconnect            1 auto      auto 1000BaseTX
.
.
port1.0.23                   disabled              1 auto      auto not present
port1.0.24                   notconnect            1 auto      auto unknown
sa1                          notconnect        trunk auto      auto
```

Table 12-2: Parameters in the output from the **show interface status** command

Parameter	Description
`Port`	Name/Type of the interface.
`Name`	Description of the interface.
`Status`	The administrative and operational status of the interface; one of: ■ disabled: the interface is administratively down. ■ connect: the interface is operationally up. ■ notconnect: the interface is operationally down.
`Vlan`	VLAN type or VLAN IDs associated with the port: ■ When the VLAN mode is trunk, it displays **trunk** (it does not display the VLAN IDs). ■ When the VLAN mode is access, it displays the VLAN ID. ■ When the VLAN mode is private promiscuous, it displays the primary VLAN ID if it has one, and **promiscuous** if it does not have a VLAN ID. ■ When the VLAN mode is private host, it displays the primary and secondary VLAN IDs. ■ When the port is an Eth port, it displays **none**: there is no VLAN associated with it. ■ When the VLAN is dynamically assigned, it displays the current dynamically assigned VLAN ID (not the access VLAN ID), or **dynamic** if it has multiple VLANs dynamically assigned.
`Duplex`	The actual duplex mode of the interface, preceded by **a-** if it has autonegotiated this duplex mode. If the port is disabled or not connected, it displays the configured duplex setting.
`Speed`	The actual link speed of the interface, preceded by **a-** if it has autonegotiated this speed. If the port is disabled or not connected, it displays the configured speed setting.
`Type`	The type of interface, e.g., 1000BaseTX. For SFP bays, it displays **Unknown** if it does not recognize the type of SFP installed, or **Not present** if an SFP is not installed or is faulty.

Related Commands show interface
show interface memory

shutdown

This command shuts down the selected interface. This administratively disables the link and takes the link down at the physical (electrical) layer.

Use the **no** variant of this command to disable this function and therefore to bring the link back up again.

Syntax

```
shutdown

no shutdown
```

Mode Interface Configuration

Examples The following example shows the use of the `shutdown` command to shut down `vlan2`.

```
         awplus# configure terminal
 awplus(config)# interface vlan2
awplus(config-if)# shutdown
```

The following example shows the use of the `no shutdown` command to bring up `vlan2`.

```
         awplus# configure terminal
 awplus(config)# interface vlan2
awplus(config-if)# no shutdown
```

Chapter 13: Interface Testing Commands

Command List

This chapter provides an alphabetical reference of commands used for testing interfaces.

clear test interface

This command clears test results and counters after issuing a test interface command. Test results and counters must be cleared to issue subsequent test interface commands later on.

Syntax `clear test interface {<port-list>|all}`

Parameter	Description
<port-list>	The ports to test. A port-list can be: ■ a switch port (e.g. `port1.0.12`) ■ a continuous range of ports separated by a hyphen, e.g. `port1.0.1-port1.0.24` ■ a comma-separated list of the above, e.g. `port1.0.1,port1.0.5-1.0.24` The specified ports must exist.
`all`	All interfaces

Mode Privileged Exec

Examples To clear the counters for `port1.0.1` use the command:

```
awplus# clear test interface port1.0.1
```

To clear the counters for all interfaces use the command:

```
awplus# clear test interface all
```

Related Commands test interface

service test

This command puts the device into the interface testing state, ready to begin testing. After entering this command, enter Interface Configuration mode for the desired interfaces and enter the command test interface.

Do not test interfaces on a device that is part of a live network—disconnect the device first.

Use the **no** variant of this command to stop the test service.

Syntax

```
service test
no service test
```

Mode Global Configuration

Example To put the device into a test state, use the command:

```
awplus(config)# service test
```

Related Commands test interface

test interface

This command starts a test on a port or all ports or a selected range or list of ports.

Use the **no** variant of this command to disable this function. The test duration can be configured by specifying the time in minutes after specifying a port or ports to test.

For an example of all the commands required to test switch ports, see the Examples section in this command. To test the Eth port, set its speed to 100 by using the command **speed 100.**

Note Do not run test interface on live networks because this will degrade network performance.

Syntax

```
test interface {<port-list>|all} [time{<1-60>|cont}]

no test interface {<port-list>|all}
```

Parameter	Description
<port-list>	The ports to test. A port-list can be: ■ a switch port (e.g. `port1.0.12`) ■ a continuous range of ports separated by a hyphen, e.g. `port1.0.1-port1.0.24` ■ a comma-separated list of the above, e.g. `port1.0.1,port1.0.5-1.0.24` The specified ports must exist.
`all`	All ports
`time`	Keyword entered prior to the value for the time duration of the interface test.
<1-60>	Specifies duration of time to test the interface or interfaces in minutes (from a minimum of 1 minute to a maximum of 60 minutes). The default is 4 minutes.
`cont`	Specifies continuous interface testing until cancelled with command negation.

Mode Privileged Exec

Example To test the switch ports in VLAN 1, install loopbacks in the ports, and enter the following commands:

```
   awplus(config)# service test

   awplus(config)# no spanning-tree rstp enable bridge-forward

   awplus(config)# interface vlan1

awplus(config-if)# shutdown

awplus(config-if)# end

          awplus# test interface all
```

To see the output, use the commands:

```
awplus# show test
```

```
awplus# show test count
```

To start the test on all interfaces for 1 minute use the command:

```
awplus# test interface all time 1
```

Related Commands clear test interface

Part 2: Layer Two Switching

- Chapter 14 Switching Introduction
- Chapter 15 Switching Commands
- Chapter 16 VLAN Introduction
- Chapter 17 VLAN Commands
- Chapter 18 Spanning Tree Introduction: STP, RSTP, and MSTP
- Chapter 19 Spanning Tree Commands
- Chapter 20 Link Aggregation Introduction and Configuration
- Chapter 21 Link Aggregation Commands
- Chapter 22 GVRP Introduction and Configuration
- Chapter 23 GVRP Commands

Chapter 14: Switching Introduction

Introduction

This chapter gives an overview of Layer 1 and 2 switching.

Layer 2 switches are used to connect multiple Local Area Network (LAN) segments together to form an extended LAN. Stations connected to different LANs can be configured to communicate with one another as if they were on the same LAN. They can also divide one physical LAN into multiple Virtual LANs (VLANs). Stations connected to each other on the same extended LAN can be grouped in separate VLANs, so that a station in one VLAN can communicate directly with other stations in the same VLAN, but must go through higher layer routing protocols to communicate with those stations in other VLANs.

Layer 2 switches appear transparent to higher layer protocols, transferring frames between the data link layers of the networks to which they are attached. A Layer 2 switch accesses each physical link according to the rules for that particular network. Access may not always be instant, so the switch must be capable of storing and forwarding frames.

Storing and forwarding enables the switch to examine both the VLAN tag fields and Ethernet MAC address fields in order to forward the frames to their appropriate destination. In this way, the switch can act as an intelligent filtering device, redirecting or blocking the movement of frames between networks.

Because switch ports can sometimes receive frames faster than it can forward them, the switch has Quality of Service (QoS) queues in which frames await transmission according to their priority. Such a situation could occur where data enters a number of input ports all destined for the same output port.

The switch can be used to:

- Increase both the physical extent and the maximum number of stations on a LAN. LANs are limited in their physical extent by the signal distortion and propagation delay characteristics of the media. The switch overcomes this limitation by receiving a frame on one LAN and then retransmitting it to another. The physical characteristics of the LAN media also place a practical limit on the number of stations that can be connected to a single LAN segment. The switch overcomes this limitation by joining LAN segments to form an extended LAN capable of supporting more stations than either of the individual LAN segments.
- Connect LANs that have a common data link layer protocol but different physical media, for example, Ethernet 10BASET, 100BASET, and 10BASEF.
- Increase the availability of LANs by allowing multiple redundant paths to be physically configured and selected dynamically, using the Spanning Tree algorithm.
- Reduce the load on a LAN or increase the effective bandwidth of a LAN, by filtering traffic.
- Prioritize the transmission of data with high Quality of Service requirements.

By using Virtual LANs (VLANs), a single physical LAN can be separated into multiple Virtual LANs. VLANs can be used to:

- Further improve LAN performance, as broadcast traffic is limited to LAN segments serving members of the VLAN to which the sender belongs.
- Provide security, as frames are forwarded to those stations belonging to the sender's VLAN, and not to stations in other VLANs on the same physical LAN.
- Reduce the cost of moving or adding stations to function or security based LANs, as this generally requires only a change in the VLAN configuration.

Physical Layer Information

Switch Ports

A unique port number identifies each switch port. The software supports a number of features at the physical level that allow it to be connected in a variety of physical networks. This physical layer (Layer 1) versatility includes:

- Enabling and disabling of ports
- Auto negotiation of port speed and duplex mode for all 10/100 BASE ports
- Manual setting of port speed and duplex mode for all 10/100 BASE ports
- Link up and link down triggers
- Packet storm protection
- Port mirroring
- Support for SNMP management

Port Numbering

Ports are numbered using a 3 digit format `x.y.z` where `x` is the device number (within a stacked configuration), `y` is the module number within the device, and `z` is the port number within the module. Ports connected directly to the switch chassis (rather than a pluggable module) are given the module number 0. In an unstacked configuration all device numbers are 1. For example, `port1.2.6` represents device 1, module 2, port 6.

Adding a description

You can add a description to an interface to help identify its purpose or position. For example, to add the description "connected to Nerv" to `port1.0.3`, use the commands:

```
   awplus(config)# interface port1.0.3
awplus(config-if)# description connected to Nerv
```

Port ranges

Continuous

To configure a continuous range of ports at the same time, enter the range in the format:

`portx.y.z-portx.y.z`

For example, to configure the same interface setting on `port1.0.10` to `port1.0.20`, enter the Global Configuration mode command:

```
awplus(config)# interface port1.0.10-port1.0.20
```

Non-continuous

To configure a non-continuous set of ports at the same time, enter a comma-separated list:

`portx.y.z,portx.y.z`

For example, to configure the same interface setting on `port1.0.1` and `port1.0.5`, enter the Global Configuration mode command:

```
awplus(config)# interface port1.0.1,port1.0.5
```

You can combine a hyphen-separated range and a comma-separated list. To configure the same setting on `port1.0.1` to `port1.0.3` and `port1.0.5`, enter the Global Configuration mode command:

```
awplus(config)# interface port1.0.1-port1.0.3,port1.0.5
```

Activating and Deactivating Switch Ports

An active switch port is one that is available for packet reception and transmission. Disabling a switch port does not affect the STP operation on the port. By default ports and VLANs are activated.

To shutdown a port or VLAN use the shutdown command on page 12.13. Use the **no** variant of this command to reactivate it.

Autonegotiation

Autonegotiation lets the port adjust its speed and duplex mode to accommodate the device connected to it. When the port connects to another autonegotiating device, they negotiate the highest possible speed and duplex mode for both of them.

By default, all ports autonegotiate. Setting the port to a fixed speed and duplex mode may be necessary when connecting to a device that cannot autonegotiate.

Duplex mode

Ports can operate in full duplex or half duplex mode depending on the type of port it is. When in full duplex mode, a port transmits and receives data simultaneously. When in half duplex mode, the port transmits or receives but not both at the same time.

You can set a port to use either of these options, or allow it to autonegotiate the duplex mode with the device at the other end of the link. To configure the duplex mode, use these commands:

`awplus#`
`configure terminal`
	Enter Global Configuration mode

`awplus(config)#`
`interface port1.0.1`
	Enter Interface Configuration mode for port 1.0.1

`awplus(config-if)#`
`duplex {auto|full|half}`
	Enter the Duplex mode for port 1.0.1

Speed options

Before configuring a port's speed, check the hardware limit for the particular port type. The following list can be used as a guide:

- non-SFP RJ-45 copper switch ports: 10, 100 or 1000 Mbps
- supported tri-speed copper SFPs: 10, 100 or 1000 Mbps
- fibre SFPs: 100 Mbps to 1000 Mbps, depending on the SFP type
- XFP modules: 10 Gbps

For the latest list of approved SFP transceivers either contact your authorized distributor or reseller, or visit http://www.alliedtelesis.com.

You can set a port to use one of these speed options, or allow it to autonegotiate the speed with the device at the other end of the link.

Most types of switch port can operate in either full duplex or half duplex mode. In full duplex mode a port can transmit and receive data simultaneously. In half duplex mode the port can either transmit or receive, but not at the same time.

Make sure that the configuration of the switch matches the configuration of the device at the far end of the link. In particular, avoid having one end autonegotiate duplex mode while the other end is fixed. For example, if you set one end of a link to autonegotiate and fix the other end at full duplex, the autonegotiating end cannot determine that the fixed end is full duplex capable. Therefore, the autonegotiating end selects half-duplex operation. This results in a duplex mismatch and packet loss. To avoid this, either fix the mode at both ends, or use autonegotiation at both ends.

Configuring the port speed

To set the port speed to 1000 kbps on port 1.0.1

awplus#	
`configure terminal`	Enter the Global Configuration mode.
awplus(config)#	
`interface port1.0.1`	Enter Interface Configuration mode for port 1.0.1
awplus(config-if)#	
`speed 1000`	Set the port speed for port 1.0.1 to 1000 Mbps.

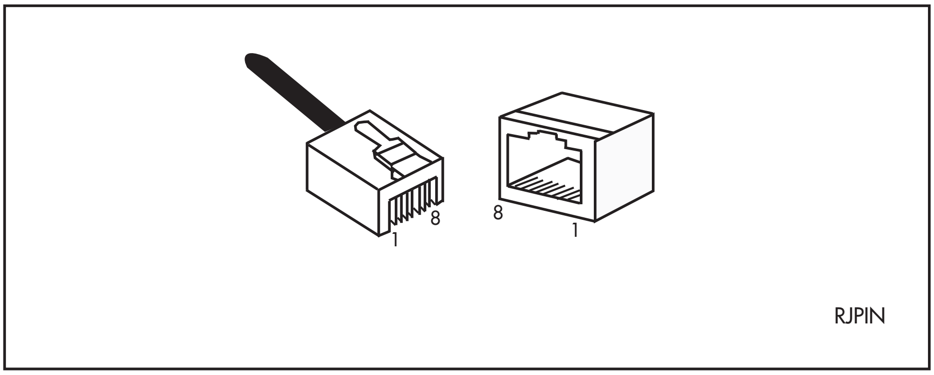
MDI/MDIX Connection Modes

By default, copper 10Base-T, 100Base-T, and 1000Base-T ports on the switch automatically set the Media Dependant Interface mode to MDI or MDIX for successful physical connections. We recommend using this default setting. However, you can configure them to have either fixed MDI mode or fixed MDIX mode by using the polarity command on page 15.25. MDI/MDIX mode polarity does not apply to fibre ports.

Connections to 10BASE-T, 100BASE-T , and 1000BASE-T networks may either be straight though (MDI) or crossover (MDIX). The crossover connection can be achieved by using either a crossover cable or by integrating the crossover function within the device. In the latter situation, the connector is referred to as an MDIX connection. Refer to your switch's Hardware Reference for more detailed information on physical connections cabling.

The IEEE 802.3 standard defines a series of Media Dependant Interface types and their physical connections. For twisted pair (10BASE-T) networking, the standard defines that connectors that conform to the IEC 60603-7 standard. The Figure 14-1f shows a connector of this type.

Figure 14-1: Connector used for 10BASE-T networks



The Layer 2 Switching Process

The Layer 2 switching process comprises these related but separate processes:

- The Ingress Rules
- The Learning Process
- The Forwarding Process
- The Egress Rules

Ingress rules admit or discard frames based on their VLAN tagging.

The Learning process learns the MAC addresses and VLAN membership of frames admitted on each port.

The Forwarding process determines which ports the frames are forwarded to, and the Quality of Service priority with which they are transmitted.

Finally, Egress rules determine for each frame whether VLAN tags are included in the Ethernet frames that are transmitted.

These processes assume that each station on the extended LAN has a unique data link layer address, and that all data link layer frames have a header which includes the source (sender's) MAC address and destination (recipient's) MAC address.

The Ingress Rules

All frames, tagged and untagged, that a VLAN-aware switch receives must be classified into a VLAN. Each received frame is mapped to exactly one VLAN. If an incoming frame is tagged with a valid VLAN identifier (VID) then that VID is used. If an incoming frame is untagged or is priority tagged (a tagged frame with a VID of all zeros), then the switch uses internal VLAN association rules to determine the VLAN it belongs to. The default settings for the ingress rules are to Admit All Frames, and for Ingress Filtering to be on.

Every port belongs to one or more VLANs so every incoming frame has a VID to show which VLAN it belongs. The final part of the Ingress Rules depends on whether Ingress Filtering is enabled for the port. If Ingress Filtering is disabled, all frames are passed on to the Learning process, regardless of which VLAN they belong to. If Ingress Filtering is enabled (by default), frame are admitted only when they have the VID of a VLAN to which the port belongs. Frames are discarded when they do not have an associated VID matching the VLAN assigned to a port.

The possible association rules, in order of precedence, are:

- IP subnet/IPX network classification
- protocol classification
- port classification

The default VLAN classification is based upon the port on which the incoming frame (untagged, or priority tagged) was received. It is possible for an incoming untagged, or priority tagged, frame to match more than one of the association rules.

Each port on the switch can be configured to be one of two modes:

- only untagged frames - access mode
- only VLAN-tagged frames - trunk mode

Access Mode

This mode can be used to connect to VLAN unaware devices. Frames to and from access mode ports carry no VLAN tagging information.

Trunk Mode

This mode is used to connect VLAN capable devices. All devices that connect using trunk mode ports must be VLAN aware.

The Learning Process

The learning process uses an adaptive learning algorithm, sometimes called ***backward learning***, to discover the location of each station on the extended LAN.

All frames admitted by the ingress rules on any port are passed on to the forwarding process when they are for destinations in the same VLAN. Frames destined for other VLANs are passed to a Layer 3 protocol, such as IP. For every frame admitted, the frame's source MAC address and VID are compared with entries in the forwarding database for the VLAN (also known as a ***MAC Address table***) maintained by the switch. When the frame's source address is not in the forwarding database for the VLAN, the address is added and an ageing timer for that entry is started. When the frame's source address is already in the forwarding database, the ageing timer for that entry is restarted.

By default, switch learning is enabled. It can be disabled with the no mac address-table acquire command, and re-enabled using the **mac address-table acquire** command on page 15.16.

If the ageing timer for an entry in the forwarding database expires before another frame with the same source address is received, the entry is removed from the forwarding database. This prevents the forwarding database from being filled with information about stations that are inactive or have been disconnected from the network. It also ensures that entries for active stations are kept alive in the forwarding database.

By default, the ageing timer is enabled with a default ageing-time. The ageing timer can be reset to the default with the **no mac address-table ageing-time** command. The ageing timer can be increased or decreased using the **mac address-table ageing-time** command.

If switch learning is disabled and the ageing timer has aged out all dynamically learned filter entries, only statically entered MAC source addresses decide the packets to forward or discard. When the switch finds no matching entries in the forwarding database during the forwarding process, all switch ports in the VLAN are flooded with the packet, except the port that received it.

The default for the mac address-table ageing-time is 300 seconds (5 minutes) and can be modified by using the command **mac address-table ageing-time**. The **no mac address-table ageing-time** command will reset the ageing-time back to the default (5 minutes).

To set the mac address-table ageing-time to 1000 seconds:

awplus# `configure terminal`	Enter the config terminal mode
awplus(config)# `mac address-table ageing-time 1000`	Set the ageing time to 1000 seconds

To display general switch settings, including settings for switch learning and the switch ageing timer, use the show system command on page 8.45.

The Forwarding Process

After a VID is assigned to a frame using the ingress rules, the switch forwards it to the destination MAC address specified in the frame. To do this the switch must learn which MAC addresses are available on each port for each VLAN. When the destination MAC address is not found, the switch floods the frame on all ports that are members of the VLAN except the port on which the frame was received.

The forwarding database (also known as the ***MAC Address table***) determines the egress port on which the destination MAC address has been learned. MAC addresses are learned dynamically as part of the Layer 2 switching process.

The forwarding database is ordered according to MAC address and VLAN identifier. This means a MAC address can appear more than once in the forwarding database having been learned on the same port but for different VLANs. This could occur if the IP address of an end station is changed thereby moving the end station to a different IP subnet-based VLAN while still connected to the same switch port. When the forwarding database ageing process is enabled, old entries in the forwarding database are deleted after a user-configurable period.

If the destination address is found, the switch discards the frame when the port is not in the STP forwarding or disabled state if the destination address is on the same port as the source address, or if there is a static filter entry for the destination address set to **discard** (see “Layer 2 Filtering” on page 14.10). Otherwise, the frame is forwarded on the indicated port.

Forwarding occurs only when the port on which the frame was received is in the Spanning Tree forwarding or disabled state. The destination address is then looked up in the forwarding database for the VLAN.

The Egress Rules

After the forwarding process has determined from which ports and transmission queues to forward a frame, the egress rules for each port determine whether the outgoing frame is VLAN-tagged with its numerical VLAN identifier (VID).

A port must belong to a VLAN at all times unless the port has been set as the mirror port for the switch.

A port can transmit VLAN-tagged frames for any VLAN to which the port belongs. A port can transmit untagged frames for any VLAN for which the port is configured, e.g. IP subnet-based or protocol-based, unless prevented by the port-based VLAN egress rules. A port that belongs to a port-based VLAN can transmit untagged packets for only one VLAN. For more information about VLANs and VLAN tagging, see Chapter 16, VLAN Introduction.

For more information on port tagging see the following commands:
switchport mode access command on page 17.14
switchport mode trunk command on page 17.20

Layer 2 Filtering

The switch has a forwarding database (also known as the ***MAC address table***) whose entries determine whether frames are forwarded or discarded over each port. Entries in the forwarding database are created dynamically by the learning process. A dynamic entry is automatically deleted from the forwarding database when its ageing timer expires.

The forwarding database supports queries by the forwarding process as to whether frames with given values of the destination MAC address field should be forwarded to a given port.

For each VLAN, the destination MAC address of a frame to be forwarded is checked against the forwarding database. If there is no entry for the destination address and VLAN, the frame is transmitted on all ports in the VLAN that are in the forwarding or disabled state, except the port on which the frame was received. This process is referred to as ***flooding***. If an entry is found in the forwarding database but the entry is not marked ***forwarding*** or the entry points to the same port the frame was received on, the frame is discarded. Otherwise, the frame is transmitted on the port specified by the forwarding database.

Ingress Filtering

The **ingress-filter** parameter of the switchport mode trunk command on page 17.20 and the switchport mode access command on page 17.14, enables or disables ingress filtering of frames entering the specified port (or port range). Each port on the switch belongs to one or more VLANs. If ingress filtering is enabled, any frame received on the specified port is only admitted if its VID matches one for which the port is tagged. Any frame received on the port is discarded if its VID does not match one for which the port is tagged.

Untagged frames are admitted and are assigned the VLAN Identifier (VID) of the port's native VLAN. Ingress filtering can be turned off by setting the **disable** parameter of the above two commands. The default setting of the **enable** / **disable** parameter option is **enable**.

Note Enabling the **vlan-disable** parameter of the thrash-limiting command on page 15.50 will also enable ingress filtering, and will override the setting of the switchport mode access, and trunk commands

Storm-control

The packet storm-control feature enables you to set limits on the reception rate of broadcast, multicast frames and destination lookup failures. You can set separate limits beyond which each of the different packet types are discarded.

Note A destination lookup failure (DLF) is the event of receiving a unicast Ethernet frame with an unknown destination address.

For more information on applying storm-control, see the **storm-control level** command on page 15.45.

Loop Protection

Loop protection is a general term that embraces several different methods you can apply to protect your network from effects such as broadcast storms that can result from data loops or equipment malfunction. Presently two methods of loop protection are available:

- Loop Detection
- Thrash Limiting

Loop Detection

Introduction

This feature is used to detect loops with a network segment. If a loop is detected then a selected protection mechanism is applied to limit the effect of the loop. The loop protection actions can be applied either to the port at which the loop is detected or to the VLAN within which the loop was detected.

Limiting Actions

You can configure loop detection to apply one of the following mechanisms when a loop condition is detected:

- Disable all MAC address learning.
- Block all traffic on the port (or aggregated link) that detected the loop, and take **down** the link.
- Block all traffic on the port (or aggregated link) that detected the loop, but keep the link in the **up** state.
- Block all traffic on a vlan. Note that setting this parameter will also enable ingress filtering. This is the default action.
- Take no action, but log the details.
- Take no action.

Operation

To detect loops this feature operates by transmitting a series of Loop Detection Frames (LDFs) from each switch port out into the network. If no loops exist, then none of these frame should ever return. If a frame returns to its original port, the detection mechanism assumes that there is a loop somewhere in the network and offers a number of protective options.

Each LDF is a Layer 2 LLC frame that contains the following components:

- the source MAC address of the originating switch
- the destination MAC address of the non-existent end station 00-00-F4-27-71-01
- VLAN ID (where the port is a tagged member of a VLAN).
- a randomly generated LDF ID number.

You can set the detection mechanism to remember the LDF ID of up to 5 of the most recently transmitted LDF frames. Each of the 5 most recently transmitted frames is compared with every frame that arrives at that same port.

Configuration

To enable loop protection and configure its basic parameters, you use the loop-protection command on page 15.13.

Example To enable the loop-detect mechanism, and generate loop-detect frames once every 5 seconds, use the command:

```
awplus(config)# loop-protection loop-detect ldf-interval 5
```

Note LDFs are sent sequentially for each VLAN defined to a particular port. For example, if a particular port in this example is a member of 4 VLANs, then the LDFs will be sent from this port at the rate of 4 frames every 5 seconds.

You can now use the loop-protection action command on page 15.14 configure the action that the switch will take if a loop is detected.

Example To disable an interface, and bring the link down, when a network loop is detected, use the command:

```
awplus(config-if)# loop-protection action link-down
```

Now decide how long you want the protective action to apply for. You configure this function by using the loop-protection timeout command on page 15.15.

Example To configure a loop protection action timeout of 10 seconds, use the command:

```
awplus(config-if)# loop-protection timeout 10
```

Thrash Limiting

MAC address thrashing occurs when MAC addresses move rapidly between one or more ports or trunks, for example, due to a network loop.

Thrash limiting enables you to apply actions to a port when thrashing is detected. It is supported on all port types and also on aggregated ports.

Limiting Actions

There are several different thrash actions that you can apply to a port when thrashing is detected. These actions are:

- learnDisable
  Address learning is temporarily disabled on the port.
- portDisable
  The port is logically disabled. Traffic flow is prevented, but the link remains up. The device at the other end does not notice that the port has changed status, and the link LEDs at both ends stay on.
- linkDown
  The port is physically disabled and the link is down. This is equivalent to entering the shutdown command on page 12.13.
- vlanDisable
  The port is disabled only for the VLAN on which thrashing has occurred. It can still receive and transmit traffic for any other VLANs of which it is a member.

When a MAC address is thrashing between two ports, one of these ports (the first to cross its thrashing threshold) is disabled. All other ports on the device will then have their threshold counters reset.

To set a thrash action for a port, use the **thrash-limiting** command on page 15.50:

To view the thrash action that is set for a port, use the **show interface switchport** command on page 15.28:

Re-enabling a port

When a port is disabled, either completely or for a specific VLAN, it remains disabled until it is manually re-enabled in any of the following ways:

- by using SNMP
- by rebooting the switch
- by specifying a thrash timeout value along with the thrash action
- via the CLI

Port Mirroring

Port mirroring enables traffic being received and transmitted on a switch port to be sent to another switch port, the mirror port, usually for the purposes of capturing the data with a protocol analyzer.

The mirror port is the only switch port that does not belong to a VLAN, and therefore does not participate in any other switching. Before the mirror port can be set, it must be removed from all trunk groups and all VLANs except the default VLAN.

The following example sets mirroring on ports 1.0.2 and 1.0.5 for both incoming and outgoing data.

Note Due to the internal hardware properties of the switch, frames that are destined to leave the mirrored port untagged (i.e. will have their VLAN tag removed on egress) will be received by the mirror port with the tag retained. Consequently, if frames were being transmitted by the mirror port (into the network) at wire speed, then the mirror port might be unable to accept all the frames supplied to it.

To configure port 1. 0. 2 to mirror port 1. 0. 5

`awplus#`
`configure terminal`
	Enter Global Configuration mode.

`awplus(config)#`
`interface port1.0.2`
	Enter the Interface Configuration mode for port1.0.2.

`awplus(config-if)#`
`mirror interface port1.0.5 direction both`
	Configure this port to mirror port 1. 0. 5.

Port Security

The port security features provide control over the stations connected to each switch port. These comprise:

- MAC address learn limits
- IEEE 802.1X

MAC Address Learn Limits

MAC address limiting is applied using the **switchport port-security** command on page 15.46. If enabled on a port, the switch will learn MAC addresses up to a user-defined limit from 1 to 256, then lock out all other MAC addresses. One of the following options can be specified for the action taken when an unknown MAC address is detected on a locked port:

- Discard the packet and take no further action.
- Discard the packet and notify management with an SNMP trap.
- Discard the packet, notify management with an SNMP trap and disable the port.

IEEE 802.1X

IEEE 802.1X restricts unauthenticated devices from connecting to the switch. After authentication is successful, traffic is allowed through the switch. For more information see Chapter 37, 802.1X Introduction and Configuration.

Quality of Service

Quality of Service (QoS) enables you to both prioritize traffic and limit its available bandwidth. The concept of QoS is a departure from the original networking protocols, in which all traffic on the Internet or within a LAN had the same available bandwidth. Without QoS, all traffic types are equally likely to be dropped if a link becomes oversubscribed. This approach is now inadequate in many networks, because traffic levels have increased and networks often carry time-critical applications such as streams of real-time video data. QoS also enables service providers to easily supply different customers with different amounts of bandwidth.

Configuring Quality of Service involves two separate stages:

1. Classifying traffic into flows, according to a wide range of criteria. Classification is performed by the switch's class maps.

2. Acting on these traffic flows.

The switch's QoS functionality includes the following:

- policies, to provide a QoS configuration for a port or ports
- traffic classes, for bandwidth limiting and user prioritization
- maximum bandwidth limiting on a traffic class
- flow groups within traffic classes, for user prioritization
- control of the egress scheduling algorithm
- priority relabelling of frames, at Layer 2, by replacing the VLAN tag User Priority field
- class of service relabelling of frames, at Layer 3, by replacing the DSCP (DiffServ Code Point) or the TOS precedence value in the IP header's Type of Service (TOS) field.

For more information on QoS see Chapter 35, Quality of Service (QoS) Introduction and Chapter 36, QoS Commands.

IGMP Snooping

IGMP (Internet Group Management Protocol) is used by IP hosts to report their multicast group memberships to routers and switches. IP hosts join a multicast group to receive broadcast messages directed to the multicast group address. IGMP is an IP-based protocol and uses IP addresses to identify both the multicast groups and the host members. For a VLAN-aware devices, this means multicast group membership is on a per-VLAN basis. If at least one port in the VLAN is a member of a multicast group, by default multicast packets will be flooded onto all ports in the VLAN.

IGMP snooping enables the switch to forward multicast traffic intelligently on the switch. The switch listens to IGMP membership reports, queries and leave messages to identify the switch ports that are members of multicast groups. Multicast traffic will only be forwarded to ports identified as members of the specific multicast group.

IGMP snooping is performed at Layer 2 on VLAN interfaces automatically. By default, the switch will forward traffic only from those ports with multicast listeners, therefore it will not act as a simple hub and flood all multicast traffic out all ports. IGMP snooping is independent of the IGMP and Layer 3 configuration, so an IP interface does not have to be attached to the VLAN, and IGMP does not have to be enabled or configured.

IGMP snooping is enabled by default.

Chapter 15: Switching Commands

Command List

This chapter provides an alphabetical reference of commands used to configure switching. For more information see Chapter 14, Switching Introduction.

backpressure

This command provides a method of applying flow control to ports running in half duplex mode. The setting will only apply when the link is in the half-duplex state.

You can disable backpressure on an interface using the **off** parameter or the **no** variant of this command.

Syntax

```
backpressure {on|off}

no backpressure
```

Parameters	Description
`on`	Enables half-duplex flow control.
`off`	Disables half-duplex flow control.

Default Backpressure is turned off by default. You can determine whether an interface has backpressure enabled by viewing the running-config output; **backpressure on** is shown for interfaces if this feature is enabled.

Mode Interface Configuration

Usage The backpressure feature enables half duplex Ethernet ports to control traffic flow during congestion by preventing further packets arriving. Back pressure utilizes a pre-802.3x mechanism in order to apply ethernet flow control to switch ports that are configured in the half duplex mode.

The flow control applied by the flowcontrol (switch port) command on page 15.11 operates only on full-duplex links, whereas back pressure operates only on half-duplex links.

If a port has insufficient capacity to receive further frames, the switch will simulate a collision by transmitting a CSMACD jamming signal from this port until the buffer empties. The jamming signal causes the sending switch to stop transmitting and wait a random period of time, before retransmitting its data, thus providing time for the buffer to clear. Although this command is only valid for switch ports operating in half-duplex mode the remote switch (the one sending the data) can be operating in the full duplex mode.

To see the currently-negotiated duplex mode for ports whose links are up, use the command show interface. To see the configured duplex mode (when different from the default), use the command **show running-config**.

Examples To enable back pressure flow control on interfaces `port1.0.1-port1.0.24` enter the following commands:

```
          awplus#  configure terminal
  awplus(config)#  interface port1.0.1-port1.0.24
awplus(config-if)#  backpressure on
```

To disable back pressure flow control on interface port1.0.2 enter the following commands:

```
         awplus# configure terminal
 awplus(config)# interface port1.0.1-port1.0.24
awplus(config-if)# backpressure off
```

Validation Commands show running-config
show interface

Related Commands duplex

clear loop-protection counters

Use this command to clear the counters for the Loop Protection counters.

Syntax `clear loop-protection [interface <port-list>] counters`

Parameters	Description
`interface`	The interface whose counters are to be cleared.
<port-list>	A port, a port range, or an aggregated link.

Mode Privileged Exec

Examples To clear the counter information:

```
awplus# clear loop-protection counters
awplus# clear loop-protection interface port1.0.1 counters
```

clear mac address-table static

Use this command to clear the filtering database of all statically configured entries for a selected MAC address, interface, or VLAN.

Syntax

```
clear mac address-table static
   [address <mac-address>|interface <port>|vlan <vid>]
```

Parameter	Description
`address`	Specify a MAC (Media Access Control) address to be cleared from the filtering database.
`<mac-address>`	Enter a MAC address to be cleared from the database in the format HHHH.HHHH.HHHH.
`interface`	Specify a switch port to be cleared from the filtering database.
`<port>`	Specify the switch port from which address entries will be cleared. This can be a single switch port, (e.g. `port1.0.4`), a static channel group (e.g. `sa3`), or a dynamic (LACP) channel group (e.g. `po4`).
`vlan`	Specify a VLAN to be cleared from the filtering database.
`<vid>`	Enter a VID (VLAN ID) in the range `<1-4094>` to be cleared from the filtering database.

Mode Privileged Exec

Usage Use this command with options to clear the filtering database of all entries made from the CLI for a given MAC address, interface or VLAN. Use this command without options to clear any entries made from the CLI.

Compare this usage with clear mac address-table dynamic command on page 15.5.

Examples This example shows how to clear all filtering database entries configured through the CLI.

```
awplus# clear mac address-table static
```

This example shows how to clear all filtering database entries for a given interface configured through the CLI.

```
awplus# clear mac address-table static interface port1.0.3
```

This example shows how to clear filtering database entries filtering database entries configured through the CLI for a given mac address.

```
awplus# clear mac address-table static address 0202.0202.0202
```

Related Commands

clear mac address-table dynamic
mac address-table static
show mac address-table

clear mac address-table dynamic

Use this command to clear the filtering database of all entries learned for a selected MAC address, an MSTP instance, a switch port interface or a VLAN interface.

Syntax

```
clear mac address-table dynamic
   [address <mac-address>|interface <port> [instance <inst>]|
   vlan <vid>]
```

Parameter	Description
`interface`	Specify a switch port to be cleared from the filtering database.
`<port>`	Specify the switch port from which address entries will be cleared. This can be a single switch port, (e.g. `port1.0.4`), a static channel group (e.g. `sa3`), or a dynamic (LACP) channel group (e.g. `po4`).
`address`	Specify a MAC (Media Access Control) address to be cleared from the filtering database.
`<mac-address>`	Enter a MAC address to be cleared from the database in the format HHHH.HHHH.HHHH.
`instance`	Specify an MSTP (Multiple Spanning Tree) instance to be cleared from the filtering database.
`<inst>`	Enter an MSTP instance in the range `<1-63>` to be cleared from the filtering database.
`vlan`	Specify a VLAN to be cleared from the filtering database.
`<vid>`	Enter a VID (VLAN ID) in the range `<1-4094>` to be cleared from the filtering database.

Mode Privileged Exec

Usage Use this command with options to clear the filtering database of all entries learned for a given MAC address, interface or VLAN. Use this command without options to clear any learned entries.

Use the optional `instance` parameter to clear the filtering database entries associated with a specified MSTP instance Note that you must first specify a switch port interface before you can specify an MSTP instance.

Compare this usage and operation with the clear mac address-table static command on page 15.4. Note that an MSTP instance cannot be specified with **clear mac address-table static**.

Examples This example shows how to clear all dynamically learned filtering database entries for all interfaces, addresses, VLANs.

```
awplus# clear mac address-table dynamic
```

This example shows how to clear all dynamically learned filtering database entries when learned through switch operation for a given MAC address.

```
awplus# clear mac address-table dynamic address 0202.0202.0202
```

This example shows how to clear all dynamically learned filtering database entries when learned through switch operation for a given MSTP instance 1 on switch port interface port1.0.2.

```
awplus# clear mac address-table dynamic interface port1.0.2
        instance 1
```

Related Commands clear mac address-table static
show mac address-table

clear port counter

Use this command to clear the packet counters of the port.

Syntax `clear port counter [<port>]`

Parameter	Description
<port>	The port number or range

Mode Privileged Exec

Example To clear the packet counter for port1.0.1

```
awplus#  clear port counter port1.0.1
```

Related Commands show platform port

debug loopprot

This command enables Loop Protection debugging.

The **no** variant of this command disables Loop Protection debugging.

Syntax

```
debug loopprot {info|msg|pkt|state|nsm|all}
```

```
no debug loopprot {info|msg|pkt|state|nsm|all}
```

Parameter	Description
`info`	General Loop Protection information.
`msg`	Received and transmitted Loop Detection Frames (LDFs).
`pkt`	Echo raw ASCII display of received and transmitted LDF packets to the console.
`state`	Loop Protection states transitions.
`nsm`	Network Service Module information.
`all`	All debugging information.

Mode Privileged Exec and Global Configuration

Example To enable debug for all state transitions, use the command:

```
awplus#  debug loopprot state
```

Related Commands show debugging loopprot
undebug loopprot

debug platform packet

This command enables platform to CPU level packet debug functionality on the switch.

Use the **no** variant of this command to disable platform to CPU level packet debug. If the result means both send and receive packet debug are disabled, then any active timeout will be cancelled.

Syntax

```
debug platform packet [recv] [send] [sflow] [timeout <timeout>]
    [vlan <vlan-id>|all]
```

```
no debug platform packet [recv] [send]
```

Parameter	Description
`recv`	Debug packets received.
`send`	Debug packets sent.
`sflow`	Debug sFlow packets.
`timeout`	Stop debug after a specified time.
<timeout>	`<0-3600>`The timeout period, specified in seconds.
`vlan`	Limit debug to a single VLAN ID specified.
<vlan-id>	`<1-4094>` The VLAN ID to limit the debug output on.
`all`	Debug all VLANs (default setting).

Default A 5 minute timeout is configured by default if no other timeout duration is specified.

Mode Privileged Exec and Global Configuration

Usage This command can be used to route packets sent and received by the CPU. If a timeout is not specified, then a default 5 minute timeout will be applied.

If a timeout of 0 is specified, packet debug will be generated until the **no** variant of this command is used or another timeout value is specified. The timeout value applies to both send and receive debug and is updated whenever the **debug platform packet** command is used.

Examples To enable both receive and send packet debug for the default timeout of 5 minutes, enter:

```
awplus# debug platform packet
```

To enable receive packet debug for 10 seconds, enter:

```
awplus# debug platform packet recv timeout 10
```

To enable packet debug for sFlow packets only for the default timeout of 5 minutes, enter:

```
awplus# debug platform packet sflow
```

To enable send packet debug with no timeout, enter:

```
awplus# debug platform packet send timeout 0
```

To enable VLAN packet debug for VLAN 2 with a timeout duration of 3 minutes, enter:

```
awplus# debug platform packet vlan 2 timeout 150
```

To disable receive packet debug, enter:

```
awplus# no debug platform packet recv
```

Related Commands show debugging platform packet
undebug platform packet

duplex

This command changes the duplex mode for the specified port.

By default, ports auto-negotiate duplex mode (except for 100Base-FX ports which do not support auto-negotiation, so default to full duplex mode).

To see the currently-negotiated duplex mode for ports whose links are up, use the command show interface. To see the configured duplex mode (when different from the default), use the command show running-config.

Syntax duplex {auto|full|half}

Parameter	Description
auto	Auto-negotiate duplex mode.
full	Operate in full duplex mode only.
half	Operate in half duplex mode only.

Mode Interface Configuration

Usage Switch ports in a static or dynamic (LACP) channel group must have the same port speed and be in full duplex mode. Once switch ports have been aggregated into a channel group, you can set the duplex mode of all the switch ports in the channel group by applying this command to the channel group.

Examples To specify full duplex for port1.0.4, enter the following commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.4
awplus(config-if)# duplex full
```

To specify half duplex for port1.0.4, enter the following commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.4
awplus(config-if)# duplex half
```

To auto-negotiate duplex mode for `port1.0.4`, enter the following commands:

```
           awplus#  configure terminal
   awplus(config)#  interface port1.0.4
awplus(config-if)#  duplex auto
```

Related Commands polarity
speed
show interface

flowcontrol (switch port)

Use this command to enable flow control, and configure the flow control mode for the switch port.

Use the **no** variant of this command to disable flow control for the specified switch port.

Syntax

```
flowcontrol both
```

```
flowcontrol {send|receive} {off|on}
```

```
no flowcontrol
```

Parameter	Description
`both`	Use this parameter to specify send and receive flow control for the port.
`receive`	When the port receives pause frames, it temporarily stops (pauses) sending traffic.
`on`	Enable the specified flow control.
`off`	Disable the specified flow control.
`send`	When the port is congested (receiving too much traffic), it sends pause frames to request the other end to temporarily stop (pause) sending traffic.

Default By default, flow control is disabled.

Mode Interface Configuration

Usage The flow control mechanism specified by 802.3x is only for full duplex links. It operates by sending PAUSE frames to the link partner to temporarily suspend transmission on the link

Flow control enables connected Ethernet ports to control traffic rates during congestion by allowing congested nodes to pause link operation at the other end. If one port experiences congestion, and cannot receive any more traffic, it notifies the other port to stop sending until the condition clears. When the local device detects congestion at its end, it notifies the remote device by sending a pause frame. On receiving a pause frame, the remote device stops sending data packets, which prevents loss of data packets during the congestion period.

Flow control is not recommended when running QoS or ACLs, because the complex queuing, scheduling, and filtering configured by QoS or ACLs may be slowed by applying flow control.

For half-duplex links, an older form of flow control known as back pressure is supported. See the related backpressure command on page 15.2.

For flow control on async serial (console) ports, see flowcontrol hardware (asyn/console) command on page 5.10.

Examples

```
          awplus#  configure terminal
  awplus(config)#  interface port1.0.2
awplus(config-if)#  flowcontrol receive on
```

```
          awplus# configure terminal
  awplus(config)# interface port1.0.2
awplus(config-if)# flowcontrol receive off
```

```
          awplus# configure terminal
  awplus(config)# interface port1.0.2
awplus(config-if)# no flowcontrol
```

Validation Commands show running-config

loop-protection

Use this command to enable the Loop Protection - loop detection - feature, and configure the detection mechanism parameters.

Use the **no** variant of this command to disable the Loop Protection feature.

Syntax

```
loop-protection loop-detect [ldf-interval <period>] [ldf-rx-window
   <frames>]
```

```
no loop-protection [loop-detect]
```

Parameter	Description
`loop-detect`	Enables loop detection when used with loop-protection keywords. Disables loop detection when used with no loop-protection keywords.
`ldf-interval`	The time (in seconds) between successive loop-detect frames being sent.
<period>	A period between 5 and 600 seconds. The default is 10 seconds.
`ldf-rx-window`	The number of transmitted loop detection frames whose details are held for comparing with frames arriving at the same port.
<frames>	A value for the window size between 1 and 5 frames. The default is 3 frames.

Default Loop Protection is disabled.

Mode Global Configuration

Usage Use this command to enable the Loop Protection feature, and configure the detection mechanism, and the detection mechanism parameters.

Example To enable the loop-detect mechanism on the switch, and generate loop-detect frames once every 5 seconds, use the command:

```
        awplus# configure terminal
awplus(config)# loop-protection loop-detect ldf-interval 5
```

loop-protection action

Use this command to specify the protective action to apply when a network loop is detected.

Use the **no** variant of this command to reset the loop protection actions to the default action, vlan-disable.

Note Currently the learn-disable parameter is not supported. If specified, an error message will be displayed.

Syntax

```
loop-protection action {learn-disable|link-down|log-only|
   port-disable|vlan-disable|none}
```

```
no loop-protection action
```

Parameter	Description
`learn-disable`	Disable MAC address learning
`link-down`	Block all traffic on a port (or aggregated link) that detected the loop, and take **down** the link.
`log-only`	Details of loop conditions are logged. No action is applied to the port (or aggregated link).
`port-disable`	Block all traffic on interface for which the loop occurred, but keep the link in the **up** state.
`vlan-disable`	Block all traffic for the VLAN on which the loop traffic was detected. Note that setting this parameter will also enable ingress filtering. This is the default action.
`none`	Applies no protective action.

Default loop-protection action vlan-disable

Mode Interface Configuration

Example To disable an interface (`port1.0.4`), and bring the link down, when a network loop is detected, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.4
awplus(config-if)# loop-protection action link-down
```

loop-protection timeout

Use this command to specify the Loop Protection recovery action duration.

Use the **no** variant of this command to set the loop protection timeout to the default.

Syntax

```
loop-protection timeout <duration>

no loop-protection timeout
```

Parameter	Description
<duration>	The time (in seconds) for which the configured action will apply before being disabled. This duration can be set between 1 and 86400 seconds (24 hours).

Default The default is 7 seconds.

Mode Interface Configuration

Example To configure a loop protection action timeout of 10 seconds for `port1.0.4`, use the command:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.4
awplus(config-if)# loop-protection timeout 10
```

mac address-table acquire

Use this command to enable MAC address learning on the device.

Use the **no** variant of this command to disable learning.

Syntax

```
mac address-table acquire

no mac address-table acquire
```

Default Learning is enabled by default for all instances.

Mode Global Configuration

Example

```
        awplus# configure terminal
awplus(config)# mac address-table acquire
```

mac address-table ageing-time

Use this command to specify an ageing-out time for a learned MAC address. The learned MAC address will persist for at least the specified time.

The **no** variant of this command will reset the ageing-out time back to the default of 300 seconds (5 minutes).

Syntax

```
mac address-table ageing-time <ageing-timer> none

no mac address-table ageing-time
```

Parameter	Description
`<ageing-timer>`	`<10-1000000>` The number of seconds of persistence.
`none`	Disable learned MAC address timeout.

Default The default ageing time is 300 seconds.

Mode Global Configuration

Examples

```
        awplus# configure terminal
awplus(config)# mac address-table ageing-time 1000
```

```
        awplus# configure terminal
awplus(config)# mac address-table ageing-time none
```

```
        awplus# configure terminal
awplus(config)# no mac address-table ageing-time
```

mac address-table static

Use this command to statically configure the MAC address-table to forward or discard frames with a matching destination MAC address.

Syntax

```
mac address-table static <mac-addr> {forward|discard} interface
    <port> [vlan <vid>]

no mac address-table static <mac-addr> {forward|discard} interface
    <port> [vlan <vid>]
```

Parameter	Description
<mac-addr>	The destination MAC address in `HHHH.HHHH.HHHH` format.
<port>	The port to display information about. The port may be a switch port (e.g. `port1.0.4`), a static channel group (e.g. `sa3`), or a dynamic (LACP) channel group (e.g. `po4`).
<vid>	The VLAN ID. If you do not specify a VLAN, its value defaults to vlan 1.

Mode Global Configuration

Usage The **mac address-table static** command is only applicable to Layer 2 switched traffic within a single VLAN. Do not apply the **mac address-table static** command to Layer 3 switched traffic passing from one VLAN to another VLAN. Frames will not be discarded across VLANs because packets are routed across VLANs. This command only works on Layer 2 traffic.

Example

```
        awplus# configure terminal
awplus(config)# mac address-table static 2222.2222.2222 forward
                interface port1.0.4 vlan 3
```

Related Commands

clear mac address-table static
show mac address-table

mac address-table thrash-limit

Use this command to set the thrash limit on the switch. Thrashing occurs when a MAC address table rapidly "flips" its mapping of a single MAC address between two subnets, usually as a result of a network loop.

Use the **no** variant of this command to disable thrash limiting.

Syntax

```
mac address-table thrash-limit <rate>

no mac address-table thrash-limit
```

Parameter	Description
<rate>	sets the maximum thrash rate at which limiting is applied. This rate can be set between 5 and 255 MAC thrashing flips per second. Once the thrash limit rate is reached, the port is considered to be thrashing.

Default No thrash limiting

Mode Global Configuration

Usage Use this command to limit thrashing on the selected port range.

Example To apply a thrash limit of 100 MAC address flips per second:

```
        awplus# configure terminal
awplus(config)# mac address-table thrash-limit 100
```

Related Commands show mac address-table thrash-limit

mirror interface

Use this command to define a mirror port and mirrored (monitored) ports and direction of traffic to be mirrored. The port for which you enter interface mode will be the mirror port.

The destination port is removed from all VLANs, and no longer participates in other switching.

Use the **no** variant of this command to disable port mirroring by the destination port on the specified source port.

Use the **none** variant of this command when using copy-to-mirror ACL and QoS commands.

Syntax

```
mirror interface <source-port-list> direction {both|receive|transmit}

mirror interface none

no mirror interface <source-port-list>

no mirror interface none
```

Parameter	Description
<source-port-list>	The source switch ports to mirror. A port-list can be: ■ a port (e.g. `port1.0.12`) ■ a continuous range of ports separated by a hyphen, e.g. `port1.0.1-1.0.24` ■ a comma-separated list of ports and port ranges, e.g. `port1.0.1,port1.0.8-1.0.24` The source port list cannot include dynamic or static channel groups (link aggregators).
`direction`	Specifies whether to mirror traffic that the source port receives, transmits, or both.
`both`	Mirroring traffic both received and transmitted by the source port.
`receive`	Mirroring traffic received by the source port.
`transmit`	Mirroring traffic transmitted by the source port.
`none`	Specify this parameter for use with the ACL (Access Control List) **access-list** and QoS (Quality of Service) default action commands when used with the **copy-to-mirror** parameter option, so you can specify the destination port (the analyzer port) for the traffic without specifying a source mirror port. See the ACL commands access-list (hardware IP numbered) and access-list (hardware MAC numbered), and the QoS command default-action for further information.

Mode Interface Configuration

Usage Use this command to send traffic to another device connected to the mirror port for monitoring.

See “Port Mirroring” on page 14.15.

A mirror port cannot be associated with a VLAN. If a switch port is configured to be a mirror port, it is automatically removed from any VLAN it was associated with.

This command can only be applied to a single mirror (destination) port, not to a range of ports, nor to a static or dynamic channel group. Do not apply multiple interfaces with an interface command before issuing the mirror interface command. One interface may have multiple mirror interfaces.

Example To mirror traffic received and transmitted on `port1.0.4` and `port1.0.5` to destination `port1.0.3`, use the commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.3
awplus(config-if)# mirror interface port1.0.4,port1.0.5
                   direction both
```

To enable use with the access-list (hardware IP numbered) ACL and default-action QoS commands to destination `port1.0.3` without specifying a source port, use the commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.3
awplus(config-if)# mirror interface none
```

Related Commands
access-list (hardware IP numbered)
access-list (hardware MAC numbered)
default-action

platform bist

This command performs a self test on the switch. This command tests the ASIC (Application Specific Integrated Circuit) memory.

Syntax

```
platform bist instance {<0-127>|all} [full]
```

Parameter	Description
`instance`	ASIC (Application Specific Integrated Circuit) instance.
`<0-127>`	ASIC instance number.
`all`	All platform instances.
`full`	Run full BIST tests.

Mode Privileged Exec

Example To run the full built in self test for all memory in the ASIC on the switch, enter the command:

```
awplus# platform bist instance all full
```

Related Commands show platform bist

platform load-balancing

This command selects which address fields are used as inputs into the load balancing algorithm. The output from this algorithm is used to select which individual path a given packet will traverse within a channel group, or aggregated link.

The **no** variant of this command applies its default setting.

Syntax

```
platform load-balancing {src-dst-mac|src-dst-ip}

no platform load-balancing
```

Parameter	Description
`src-dst-mac`	Include the Source and Destination MAC addresses (Layer 2)
`src-dst-ip`	Include the Source and Destination IP addresses (Layer 3)

Default Includes the **src-dst-mac** and **src-dst-ip** address components into the platform load balancing algorithm.

Mode Global Configuration

Examples To set the load balancing algorithm to include Layer 2 MAC addresses, enter:

```
        awplus# configure terminal
awplus(config)# platform load-balancing src-dst-mac
```

To set the load balancing algorithm to include Layer 3 IP addresses, enter:

```
        awplus# configure terminal
awplus(config)# platform load-balancing src-dst-ip
```

Related Commands show platform

platform vlan-stacking-tpid

This command specifies the Tag Protocol Identifier (TPID) value that applies to all frames that are carrying double tagged VLANs. All nested VLANs must use the same TPID value. (This feature is sometimes referred to as VLAN stacking or VLAN double-tagging.)

Use the **no** variant of this command to revert to the default TPID value (0x8100).

Syntax

```
platform vlan-stacking-tpid <tpid>
```

```
no platform vlan-stacking-tpid
```

Parameter	Description
<tpid>	The Ethernet type of the tagged packet, as a two byte hexadecimal number.

Default The default TPID value of 0x8100 is restored using a **no platform vlan-stacking-tpid** command.

Mode Global Configuration

Examples To set the VLAN stacking TPID value to 0x9100, use the following commands:

```
        awplus# configure terminal
awplus(config)# platform vlan-stacking-tpid 9100
```

To reset the VLAN stacking TPID value to the default (0x8100), use the following commands:

```
        awplus# configure terminal
awplus(config)# no platform vlan-stacking-tpid
```

Related Commands show platform
show running-config

polarity

This command sets the MDI/MDIX polarity on a copper-based switch port.

Syntax

```
polarity {auto|mdi|mdix}
```

Parameter	Description
`mdi`	Sets the polarity to MDI (medium dependent interface).
`mdix`	Sets the polarity to MDI-X (medium dependent interface crossover).
`auto`	The switch port sets the polarity automatically. This is the default option.

Default By default, switch ports set the polarity automatically (**auto**).

Mode Interface Configuration

Usage We recommend the default **auto** setting for MDI/MDIX polarity. Polarity applies to copper 10BASE-T, 100BASE-T, and 1000BASE-T switch ports; It does not apply to fibre ports. For more information, see "MDI/MDIX Connection Modes" on page 14.5.

Example

show debugging loopprot

This command shows Loop Protection debugging information.

Syntax

```
show debugging loopprot
```

Mode User Exec and Privileged Exec

Example To display the enabled Loop Protection debugging modes, use the command:

```
awplus# show debugging loopprot
```

Related Commands debug loopprot

show debugging platform packet

This command shows platform to CPU level packet debugging information.

Syntax

```
show debugging platform packet
```

Mode

User Exec and Privileged Exec

Example

To display the platform packet debugging information, use the command:

```
awplus# show debugging platform packet
```

Related Commands

debug platform packet
undebug platform packet

show flowcontrol interface

Use this command to display flow control information.

Syntax `show flowcontrol interface <port>`

Parameter	Description
<port>	Specifies the name of the port to be displayed.

Mode User Exec and Privileged Exec

Example To display the flow control for the `port1.0.5`, use the command:

```
awplus# show flowcontrol interface port1.0.5
```

Output Figure 15-1: Example output from the **show flowcontrol interface** command for a specific interface

```
Port     Send  FlowControl    Receive  FlowControl  RxPause TxPause
          admin    oper         admin    oper
-----     ------- --------     ------- --------     ------- -------
port1.0.5 on        on            on        on             0       0
```

show interface switchport

Use this command to show VLAN information about each switch port.

Syntax `show interface switchport`

Mode User Exec and Privileged Exec

Example To display VLAN information about each switch port, enter the command:

```
awplus# show interface switchport
```

Output Figure 15-2: Example output from the **show interface switchport** command

```
 Interface name         : port1.0.1
 Switchport mode        : access
 Ingress filter         : enable
 Acceptable frame types : all
 Default Vlan           : 2
 Configured Vlans       : 2

 Interface name         : port1.0.2
 Switchport mode        : trunk
 Ingress filter         : enable
 Acceptable frame types : all
 Default Vlan           :  1
 Configured Vlans       : 1 4 5 6 7 8
...
```

Related Commands show interface memory

show loop-protection

Use this command to display the current loop protection setup for the device.

Syntax `show loop-protection [interface <port-list>] [counters]`

Parameter	Description
`interface`	The interface selected for display.
`<port-list>`	A port, a port range, or an aggregated link.
`counters`	Displays counter information for loop protection.

Mode User Exec and Privileged Exec

Usage This command is used to display the current configuration and operation of the Loop Protection feature

Examples To display the current configuration status for `port1.0.1`, use the command:

```
awplus# show loop-protection interface port1.0.1
```

Figure 15-3: Example output from the **show loop-protection** command

```
Loop-Detection:          Enabled
LDF Interval:            10 [sec]
Interface:               port1.0.1
 Action:                 port-disable
 Timeout:                300 [sec]
 Vlan:                   1
  Status:                Blocking
  Timeout Remaining:     115 [sec]
 Vlan:                   2
  Status:                Normal
  Timeout Remaining:     0 [sec]
```

To display the counter information for `port1.0.1`, use the command:

```
awplus# show loop-protection interface port1.0.1 counters
```

Figure 15-4: Example output from the **show loop-protection interface counters** command for port1.0.1

```
Interface:               port1.0.1
 Vlan:                   1
  LDF Tx:                3
  LDF Rx:                1
  Invalid LDF Rx:        1
  Action:                1
Vlan:                    2
  LDF Tx:                3
  LDF Rx:                0
  Invalid LDF Rx:        0
  Action:                0
```

show mac address-table

Use this command to display the mac address-table for all configured VLANs.

Syntax

```
show mac address-table
```

Mode User Exec and Privileged Exec

Usage The **show mac address-table** command is only applicable to view a mac address-table for Layer 2 switched traffic within VLANs.

Example To display the mac address-table, use the following command:

```
awplus# show mac address-table
```

Output See the below sample output captured when there was no traffic being switched:

```
awplus#show mac address-table

VLAN Port          MAC              State
1    unknown       0000.cd28.0752   static
ARP  -             0000.cd00.0000   static
```

See the sample output captured when packets were switched and mac addresses were learnt:

```
awplus#show mac address-table

VLAN Port          MAC              State
1    unknown       0000.cd28.0752   static
1    port1.0.11    0030.846e.9bf4   dynamic
1    port1.0.9     0030.846e.bac7   dynamic
ARP  -             0000.cd00.0000   static
```

Note the new mac addresses learnt for `port1.0.9` and `port1.0.11` added as dynamic entries.

Note the first column of the output below shows VLAN IDs if multiple VLANs are configured:

```
awplus#show mac address-table

VLAN Port          MAC              State
1    unknown       0000.cd28.0752   static
1    port1.0.9     0030.846e.bac7   dynamic
2    unknown       0000.cd28.0752   static
2    port1.0.11    0030.846e.9bf4   dynamic
ARP  -             0000.cd00.0000   static
```

Also note manually configured static mac-addresses are shown to the right of the type column:

```
awplus(config)#mac address-table static 0000.1111.2222 for int
port1.0.11 vlan 2
awplus(config)#end
awplus#
awplus#show mac address-table

VLAN Port         MAC               State
1    unknown      0000.cd28.0752    static
1    port1.0.9    0030.846e.bac7    dynamic
2    port1.0.11   0000.1111.2222    static
2    unknown      0000.cd28.0752    static
2    port1.0.11   0030.846e.9bf4    dynamic
ARP  -            0000.cd00.0000    statics
```

Related Commands clear mac address-table dynamic
clear mac address-table static
mac address-table static

show mac address-table thrash-limit

Use this command to display the current thrash limit set for all interfaces on the device.

Syntax

```
show mac address-table thrash-limit
```

Mode User Exec and Privileged Exec

Example To display the current, use the following command:

```
awplus# show mac address-table thrash-limit
```

Output Figure 15-5: Example output from the **show mac address-table thrash-limit** command

```
% Thrash-limit  7 movements per second
```

Related Commands mac address-table thrash-limit

show mirror

Use this command to display the status of all mirrored ports.

Syntax `show mirror`

Mode User Exec and Privileged Exec

Example To display the status of all mirrored ports, use the following command:

```
awplus# show mirror
```

Output Figure 15-6: Example output from the **show mirror** command

```
Mirror Test Port Name: port1.0.1
Mirror option: Enabled
Mirror direction: both
Monitored Port Name: port1.0.2
Mirror Test Port Name: port1.0.3
Mirror option: Enabled
Mirror direction: receive
Monitored Port Name: port1.0.4
Mirror Test Port Name: port1.0.3
Mirror option: Enabled
Mirror direction: receive
Monitored Port Name: port1.0.1
Mirror Test Port Name: port1.0.1
Mirror option: Enabled
Mirror direction: receive
Monitored Port Name: port1.0.3
Mirror Test Port Name: port1.0.1
Mirror option: Enabled
Mirror direction: transmit
Monitored Port Name: port1.0.4
```

show mirror interface

Use this command to display port mirroring configuration for a mirrored (monitored) switch port.

Syntax

```
show mirror interface <port>
```

Parameter	Description
<port>	The monitored switch port to display information about.

Mode User Exec, Privileged Exec and Interface Configuration

Example To display port mirroring configuration for the `port1.0.4`, use the following commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.4
awplus(config-if)# show mirror interface port1.0.4
```

Output Figure 15-7: Example output from the **show mirror interface** command

```
Mirror Test Port Name: port1.0.3
Mirror option: Enabled
Mirror direction: both
Monitored Port Name: port1.0.4
```

show platform

This command displays the settings configured by using the **platform** commands.

Syntax

```
show platform
```

Mode Privileged Exec

Usage This command displays the settings in the running config. For changes in some of these settings to take effect, the switch must be rebooted with the new settings in the startup config.

Example To check the settings configured with **platform** commands on the switch, use the following command:

```
awplus# show platform
```

Output Figure 15-8: Example output from the **show platform** command

```
awplus# show platform

Vlan-stacking TPID             0x8100
```

Table 15-1: Parameters in the output of the **show platform** command

Parameter	Description
Vlan-stacking TPID	The value of the TPID set in the Ethernet type field when a frame has a double VLAN tag (platform vlan-stacking-tpid command on page 15.24).

Related Commands platform vlan-stacking-tpid

show platform bist

This command displays the result of a previously run BIST (Built In Self Test) on the switch.

Syntax

```
show platform bist
```

Mode Privileged Exec

Example To show the result of a previously run BIST on the switch, enter the following command:

```
awplus# show platform bist
```

Output Figure 15-9: Example output from the **show platform bist** command

```
Platform Built In Self Test Results
  Switch Instance 0 ......... Passed
00   forward   static
```

Related Commands platform bist

show platform classifier statistics utilization brief

This command displays the total memory space, and free memory space of CAM (Content-Addressable Memory). Utilization statistics for various platform functions, such as ACLs and QoS are also shown.

Syntax `show platform classifier statistics utilization brief`

Mode Privileged Exec

Example To display the platform classifier utilization statistics, use the following command:

```
awplus# show platform classifier statistics utilization brief
```

Output Figure 15-10: Output from the **show platform classifier statistics utilization brief** command, with the DOS detection feature disabled.

```
[Instance 3.0]
(Port1.0.1-1.0.24)
  Number of Entries:
  Policy Type      Group ID   Used / Total
  ----------------------------------------------
  ACL              1476395009    0 /  122 (  0%)
  DoS                      -1    0 /    0 (  0%)
  VLAN Counter             -1    0 /    0 (  0%)
  QoS                            0 /  768 (  0%)

[Instance 3.1]
(Port1.0.25-1.0.48)
  Number of Entries:
  Policy Type      Group ID   Used / Total
  ----------------------------------------------
  ACL              1476395009    0 /  122 (  0%)
  DoS                      -1    0 /    0 (  0%)
  VLAN Counter             -1    0 /    0 (  0%)
  QoS                            2 /  768 (  0%)
```

Figure 15-11: Output from the **show platform classifier statistics utilization brief** command, with the DOS detection feature enabled.

```
[Instance 3.0]
[Port1.0.1-1.0.24]
Number of Entries:
  Policy Type      Group ID   Used / Total
  ----------------------------------------------
  ACL              1476395009    0 /  122 (  0%)
  DoS              1476395011    0 /  128 (  0%)
  VLAN Counter             -1    0 /    0 (  0%)
  QoS                            0 /  640 (  0%)

[Instance 3.1]
[Port1.0.25-1.0.48]
  Number of Entries:
  Policy Type      Group ID   Used / Total
  ----------------------------------------------
  ACL              1476395009    0 /  122 (  0%)
  DoS              1476395011    1 /  128 (  0%)
  VLAN Counter             -1    0 /    0 (  0%)
  QoS                            2 /  640 (  0%)
                            1    2 /  128 (  1%)
```

show platform port

This command displays the various port registers or platform counters for specified switchports.

Syntax `show platform port [<port-list>|counters]`

Parameter	Description
<port-list>	The ports to display information about. A port-list can be: ■ a continuous range of ports separated by a hyphen, e.g. `port1.0.1-1.0.24` ■ a comma-separated list of ports and port ranges, e.g. `port1.0.1,port1.0.7-1.0.24`.
`counters`	Show the platform counters.

Mode Privileged Exec

Examples To display port registers for `port1.0.1` and `port1.0.2` use the following command:

```
awplus# show platform port port1.0.1-port1.0.2
```

To display platform counters for `port1.0.1` and `port1.0.2` use the following command:

```
awplus# show platform port port1.0.1-port1.0.2 counters
```

Output Figure 15-12: Example output from the **show platform port** command

```
awplus#show platform port port1.0.1
Phy register value for port1.0.1 (ifindex: 5001)

00:1140  01:7949  02:0020  03:60B1  04:01E1  05:0000  06:0004  07:2001
08:0000  09:0600  10:0000  11:0000  12:0000  13:0000  14:0000  15:0000
16:0000  17:0000  18:0000  19:0000  20:0000  21:0000  22:0000  23:0000
24:0000  25:0000  26:0000  27:0000  28:0000  29:0000  30:0000  31:0000

Port configuration for lport 0x08001000:
  enabled:              1
  loopback:             0
  link:                 0
  speed:                0  max speed:            1000
  duplex:               0
  linkscan:             2
  autonegotiate:        1
  master:               2
  tx pause:             1  rx pause:                1
  untagged vlan:        1
  vlan filter:          3
  stp state:            1
  learn:                5
  discard:              0
  max frame size:    1522
  MC Disable SA:       no
  MC Disable TTL:      no
  MC egress untag:      0
  MC egress vid:        0
  MC TTL threshold:    -1
```

Output Figure 15-13: Example output from the **show platform port counters** command

```
awplus#show platform port port1.0.1 counters

 Switch Port Counters
--------------------------------------------------------------------------

Port port1.0.1 Ethernet MAC counters:
 Combined receive/transmit packets by size (octets) counters:
  64                               0 1024 - MaxPktSz                  0
  65 - 127                         0 1519 - 1522                      0
  128 - 255                        0 1519 - 2047                      0
  256 - 511                        0 2048 - 4095                      0
  512 - 1023                       0 4096 - 9216                      0

 General Counters:
 Receive                             Transmit
  Octets                           0 Octets                           0
  Pkts                             0 Pkts                             0
  FCSErrors                        0
  UnicastPkts                      0 UnicastPkts                      0
  MulticastPkts                    0 MulticastPkts                    0
  BroadcastPkts                    0 BroadcastPkts                    0
  PauseMACCtlFrms                  0 PauseMACCtlFrms                  0
  OversizePkts                     0
  Fragments                        0
  Jabbers                          0
  UnsupportOpcode                  0
  AlignmentErrors                  0
  SymErDurCarrier                  0
  CarrierSenseErr                  0
  UndersizePkts                    0
                                     FrameWDeferrdTx                  0
                                     FrmWExcesDefer                   0
                                     SingleCollsnFrm                  0
                                     MultCollsnFrm                    0
                                     LateCollisions                   0
                                     ExcessivCollsns                  0
                                     Collisions                       0

 Layer 3 Counters:
  ifInUcastPkts                    0 ifOutUcastPkts                   0
  ifInDiscards                     0 ifOutErrors                      0
  ipInHdrErrors                    0

 Miscellaneous Counters:
  DropEvents                       0
  ifOutDiscards                    0
  MTUExcdDiscard                   0

--------------------------------------------------------------------------
```

show port-security interface

Use this command to show the current port-security configuration and the switch port status.

Syntax `show port-security interface <port>`

Parameter	Description
`<port>`	The port to display information about. The port may be a switch port (e.g. `port1.0.4`), a static channel group (e.g. `sa3`), or a dynamic (LACP) channel group (e.g. `po4`).

Mode Privileged Exec

Example To see the port-security status on `port1.0.1`, use the following command:

```
awplus# show port-security interface port1.0.1
```

Output Figure 15-14: Example output from the **show port-security interface** command

```
Port Security configuration
Security Enabled                : YES
Port Status                     : ENABLED
Violation Mode                  : TRAP
Aging                           : OFF
Maximum MAC Addresses           : 3
Total MAC ddresses              : 1
Lock Status                     : UNLOCKED
Security Violation Count        : 0
Last Violation Source Address : None
```

show port-security intrusion

Shows the intrusion list. If the port is not give, entire intrusion table is shown.

Syntax

```
show port-security intrusion [interface <port>]
```

Parameter	Description
`interface`	Specify a port
`<port>`	The port to display information about. The port may be a switch port (e.g. `port1.0.4`), a static channel group (e.g. `sa3`), or a dynamic (LACP) channel group (e.g. `po4`).

Mode Privileged Exec

Example To see the intrusion list on `port1.0.1`, use the following command:

```
awplus# show port-security intrusion interface port1.0.1
```

Output Figure 15-15: Example output from the **show port-security intrusion** command for port 1.0.1

```
Port Security Intrusion List
Interface: port1.0.1 -3 intrusion(s) detected
11-22-33-44-55-04 11-22-33-44-55-06 11-22-33-44-55-08
```

show storm-control

Use this command to display storm-control information for all interfaces or a particular interface.

Syntax `show storm-control [<port>]`

Parameter	Description
<port>	The port to display information about. The port may be a switch port (e.g. `port1.0.4`), a static channel group (e.g. `sa3`), or a dynamic (LACP) channel group (e.g. `po4`).

Mode User Exec and Privileged Exec

Example To display storm-control information for `port1.0.2`, use the following command:

```
awplus# show storm-control port1.0.2
```

Output Figure 15-16: Example output from the **show storm-contro**l command for port1.0.2

```
Port        BcastLevel  McastLevel  DlfLevel
port1.0.2       40. 0%      100. 0%   100. 0%
```

Example To display storm-control information for all ports, use the following command:

```
awplus# show storm-control
```

Output Figure 15-17: Example output from the **show storm-control** command for all ports

```
awplus#show storm-control
Port        BcastLevel  McastLevel  DlfLevel
port1.0.1       100.0%      100.0%    100.0%
port1.0.2       100.0%      100.0%    100.0%
port1.0.3       100.0%      100.0%    100.0%
port1.0.4       100.0%      100.0%    100.0%
port1.0.5       100.0%      100.0%    100.0%
port1.0.6       100.0%      100.0%    100.0%
port1.0.7       100.0%      100.0%    100.0%
port1.0.8       100.0%      100.0%    100.0%
port1.0.9       100.0%      100.0%    100.0%
port1.0.10      100.0%      100.0%    100.0%
port1.0.11      100.0%      100.0%    100.0%
port1.0.12      100.0%      100.0%    100.0%
port1.0.13      100.0%      100.0%    100.0%
port1.0.14      100.0%      100.0%    100.0%
port1.0.15      100.0%      100.0%    100.0%
port1.0.16      100.0%      100.0%    100.0%
port1.0.17      100.0%      100.0%    100.0%
port1.0.18      100.0%      100.0%    100.0%
port1.0.19      100.0%      100.0%    100.0%
port1.0.20      100.0%      100.0%    100.0%
port1.0.21      100.0%      100.0%    100.0%
port1.0.22      100.0%      100.0%    100.0%
port1.0.23      100.0%      100.0%    100.0%
port1.0.24      100.0%      100.0%    100.0%
```

Related Commands storm-control level

speed

This command changes the speed of the specified port. You can optionally specify the speed or speeds that get autonegotiated, so autonegotiation is only attempted at the specified speeds.

To see the currently-negotiated speed for ports whose links are up, use the show interface command. To see the configured speed (when different from the default), use the show running-config command.

Syntax

```
speed {10|100|1000|10000|auto [10][100][1000][10000]}
```

The following table shows the speed options for each type of port.

Port type	Speed Options (units are Mbps)
non-SFP RJ-45 copper ports	auto (default) 10 100 1000
supported tri-speed copper SFPs	auto (default) 10 100 1000
100Mb fibre SFPs	100
1000Mb fibre SFPs	auto (default) 1000

Mode Interface Configuration

Default By default, ports autonegotiate speed (except for 100Base-FX ports which do not support auto-negotiation, so default to 100Mbps).

Usage Switch ports in a static or dynamic (LACP) channel group must have the same port speed and be in full duplex mode. Once switch ports have been aggregated into a channel group, you can set the speed of all the switch ports in the channel group by applying this command to the channel group.

Note Note that if multiple speeds are specified after the auto option to autonegotiate speeds, then only those speeds specified are attempted for autonegotiation.

Examples To set the speed of a tri-speed port to 100Mbps, enter the following commands:

```
          awplus#  configure terminal
  awplus(config)#  interface port1.0.4
awplus(config-if)#  speed 100
```

To return the port to auto-negotiating its speed, enter the following commands:

```
          awplus#  configure terminal
  awplus(config)#  interface port1.0.4
awplus(config-if)#  speed auto
```

To set a port to auto-negotiate its speed at 100Mbps and 1000Mbps, enter the following commands:

```
          awplus#  configure terminal
  awplus(config)#  interface port1.0.2
awplus(config-if)#  speed auto 100 1000
```

To set a port to auto-negotiate its speed at 1000Mbps only, enter the following commands:

```
          awplus#  configure terminal
  awplus(config)#  interface port1.0.2
awplus(config-if)#  speed auto 1000
```

Related Commands duplex
polarity
show interface
speed (asyn)

storm-control level

Use this command to specify the threshold level for broadcasting, multicast, or destination lookup failure (DLF) traffic for the port. Storm-control limits the specified traffic type to the specified threshold.

Use the **no** variant of this command to disable storm-control for broadcast, multicast or DLF traffic.

Syntax

```
storm-control {broadcast|multicast|dlf} level <level>
```

```
no storm-control {broadcast|multicast|dlf} level
```

Parameter	Description
<level>	`<0-100>` Specifies the threshold as a percentage of the maximum port speed.
`broadcast`	Applies the storm-control to broadcast frames.
`multicast`	Applies the storm-control to multicast frames.
`dlf`	Applies the storm-control to destination lookup failure traffic.

Default By default, storm-control is disabled.

Mode Interface Configuration

Usage Flooding techniques are used to block the forwarding of unnecessary flooded traffic. A packet storm occurs when a large number of broadcast packets are received on a port. Forwarding these packets can cause the network to slow down or time out.

Example

Related Commands show storm-control

switchport port-security

Enables the port-security feature. This feature is also known as port-based learn limit. It allows the user to set the maximum number of MAC addresses that each port can learn.

Use the **no** variant of this command to disable the port-security feature.

Syntax

```
switchport port-security
```

```
no switchport port-security
```

Mode Interface Configuration

Examples To enable the port-security feature on port1.0.4, use the following commands:

```
         awplus# configure terminal
 awplus(config)# interface port1.0.4
awplus(config-if)# switchport port-security
```

To disable port-security feature on port1.0.4, use the following commands:

```
         awplus# configure terminal
 awplus(config)# interface port1.0.4
awplus(config-if)# no switchport port-security
```

switchport port-security aging

Sets the port-security MAC to time out.

Use the **no** variant of this command to set the port-security to not time out.

Syntax

```
switchport port-security aging
```

```
no switchport port-security aging
```

Mode Interface Configuration

Examples To set the MAC to time out, use the following command:

```
awplus# switchport port-security aging
```

To unset the MAC time out, use the following command:

```
awplus# no switchport port-security aging
```

switchport port-security maximum

Sets the maximum MAC address that each port can learn.

Use the **no** variant of this command to unset the maximum number of MAC addresses that each port can learn. This is same as setting the maximum number to 0. This command also resets the intrusion list table.

Syntax

```
switchport port-security maximum <0-256>
```

```
no switchport port-security maximum
```

Parameter	Description
`maximum`	Maximum number of address to learn
<0-256>	Maximum number of address to learn

Mode Interface Configuration

ExamplesTo learn 3 MAC addresses on `port1.0.4`, use the following commands:

```
          awplus#  configure terminal
  awplus(config)#  interface port1.0.4
awplus(config-if)# switchport port-security maximum 3
```

To remove the MAC learning limit on `port1.0.4`, use the following commands:

```
          awplus#  configure terminal
  awplus(config)#  interface port1.0.4
awplus(config-if)# no switchport port-security maximum
```

switchport port-security violation

Sets the violation action for a switch port when the port exceeds the learning limits. The port action can be either **shutdown**, **restrict** or **protect**. If **shutdown** is set, the physical link will be disabled and "shutdown" will be shown in the config. If **restrict** is set, the packet from the un-authorized MAC will be discarded and SNMP TRAP will be generated to alert management. If **protect** is set, the packet will simply be discarded by the packet processor silently.

The **no** variant of this command sets the violation action to default. The default violation action is protect.

Syntax

```
switchport port-security violation {shutdown|restrict|protect}
```

```
no switchport port-security violation
```

Parameter	Description
`shutdown`	Disable the port
`restrict`	Alert the network administrator
`protect`	Discard the packet

Mode Interface Configuration

Examples To set the action to be shutdown on `port1.0.4`, use the following commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.4
awplus(config-if)# switchport port-security violation shutdown
```

To set the port-security action to the default (protect) on `port1.0.4`, use the following commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.4
awplus(config-if)# no switchport port-security violation
```

thrash-limiting

Sets and configures the thrash limit action that will be applied to any port on the switch when a thrashing condition is detected. The thrash-limiting timeout specifies the time, in seconds, for which the thrash action is employed.

Syntax

```
thrash-limiting {[action {learn-disable|link-down|port-disable|
   vlan-disable|none}] [timeout <0-86400>]}
```

```
no thrash-limiting {action|timeout}
```

Parameter	Description
`action`	The mac thrashing detected action. The default is vlan-disable.
`learn-disable`	Disable mac address learning
`link-down`	Block all traffic on an interface - link down
`port-disable`	Block all traffic on an interface - link remains up
`vlan-disable`	Block all traffic on a vlan Note that setting this parameter will also enable ingress filtering.
`none`	No thrash action
`timeout`	Set the duration for the thrash action
<0-86400>	The duration of the applied thrash action in seconds. The default is 1 seconds.

Default The default action is vlan-disable.

Mode Interface Configuration

Examples To set the action to learn disable for `port1.0.4`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.4
awplus(config-if)# thrash-limiting action learn-disable
```

To set the thrash limiting timeout to 5 seconds, use the following command:

```
awplus(config-if)# thrash-limiting timeout 5
```

To set the thrash limiting action to its default, use the following command:

```
awplus(config-if)# no thrash-limiting action
```

To set the thrash limiting timeout to its default, use the following command:

```
awplus(config-if)# no thrash-limiting timeout
```

undebug platform packet

This command applies the functionality of the no debug platform packet command on page 15.8.

undebug loopprot

This command applies the functionality of the no debug loopprot command on page 15.7.

Chapter 16: VLAN Introduction

Introduction

This chapter describes Virtual LANs (VLAN), VLAN features and configuration on the switch. For detailed descriptions of commands used to configure VLANs, see Chapter 17, VLAN Commands. For information about Voice VLAN and LLDP-MED, see Chapter 65, LLDP Introduction and Configuration.

Virtual LANs (VLANs)

A Virtual LAN (VLAN) is a logical, software-defined subnetwork. It allows similar devices on the network to be grouped together into one broadcast domain, irrespective of their physical position in the network. Multiple VLANs can be used to group workstations, servers, and other network equipment connected to the switch, according to similar data and security requirements.

Decoupling logical broadcast domains from the physical wiring topology offers several advantages, including the ability to:

- Move devices and people with minimal, or no, reconfiguration
- Change a device's broadcast domain and access to resources without physically moving the device, by software reconfiguration or by moving its cable from one switch port to another
- Isolate parts of the network from other parts, by placing them in different VLANs
- Share servers and other network resources without losing data isolation or security
- Direct broadcast traffic to only those devices which need to receive it, to reduce traffic across the network
- Connect 802.1Q-compatible switches together through one port on each switch

Devices that are members of the same VLAN only exchange data with each other through the switch's Layer 2 switching capabilities. To exchange data between devices that are located in different VLANs, the switch's Layer 3 (routing) capabilities are used.

Different IP subnets are associated with different VLANs. The switch's IP router table will be populated by the routes to the subnets on any active VLANs, and by routes statically configured over active VLAN interfaces, or learnt via routing protocols operating over these interfaces.

The device supports up to 4094 VLANs (the maximum allowed by the VID field in the 802.1Q tag). On some devices a few of these VLANs may be reserved for management purposes.

When the switch is first powered up (and therefore unconfigured), it creates a default VLAN with a VID of 1 and an interface name of *vlan1*. In this initial condition, the switch attaches all its ports to this default VLAN.

The default VLAN cannot be deleted, and ports can only be removed from it if they also belong to at least one other VLAN. If all the devices on the physical LAN belong to the same logical LAN, that is, the same broadcast domain, then the default settings will be acceptable, and no additional VLAN configuration is required.

Configuring VLANs

Defaults By default, all switch ports are in access mode, are associated with the default VLAN (**vlan1**), and have ingress filtering on. You cannot delete **vlan1**.

VLAN names When you create a VLAN (using the vlan command), you give it a numerical VLAN Identifier (VID) - a number from 2 to 4094. If tagged frames are transmitted from this VLAN, they will contain this VID in their tag. You may also give it an arbitrary alphanumeric name containing a meaningful description, which is not transmitted to other devices.

When referring to a VLAN, some commands require the VLAN to be specified by its VID while some commands require it to be specified by its interface name: vlan<VID>. In command output, the VLAN may be referred to by its VID, its interface name (vlan<VID>), or its VLAN name (the arbitrary alphanumeric string).

You can name a VLAN with a string containing “vlan” and its VLAN Identifier (VID). To avoid confusion, we recommend not naming it “vlan” followed by any number different from its VID.

Access mode A switch port in access mode sends untagged Ethernet frames, that is, frames without a VLAN tag. Each port is associated with one VLAN (the port-based VLAN, by default, *vlan1*), and when it receives untagged frames, it associates them with the VID of this VLAN. You can associate the port with another VLAN (using the switchport access vlan command). This removes it from the default VLAN.

Use access mode for any ports connected to devices that do not use VLAN tagging, for instance PC workstations.

Trunk mode A switch port in trunk mode is associated with one or more VLANs for which it transmits VLAN-tagged frames, and for which it identifies incoming tagged frames with these VIDs.

To allow a switch port to distinguish and identify traffic from different VLANs, put it in trunk mode (using the switchport mode trunk command), and add the VLANs (using the switchport trunk allowed vlan command). Use trunk mode for ports connected to other switches which send VLAN-tagged traffic from one or more VLANs.

A trunk mode port may also have a native VLAN (by default vlan1), for which it transmits untagged frames, and with which it associates incoming untagged frames (using the switchport trunk native vlan command).

Ports in trunk mode can be enabled as promiscuous ports for private VLANs (using the switchport mode private-vlan trunk promiscuous) and secondary ports for private VLANs (using the switchport mode private-vlan trunk secondary).

Mirror ports A mirror port cannot be associated with a VLAN. If a switch port is configured to be a mirror port (using the mirror interface command), it is automatically removed from any VLAN it was associated with.

VLANs and channel groups All the ports in a channel group must have the same VLAN configuration: they must belong to the same VLANs and have the same tagging status, and can only be operated on as a group.

Table 16-1: Configuration procedure for VLANs

Create VLANs	
awplus# configure terminal	Enter Configuration mode.
awplus(config)# vlan database	Enter VLAN Configuration mode.
awplus(config-vlan)# vlan <vid> [name <vlan-name>] [state {enable|disable}] or vlan <vid-range> [state {enable|disable}]	Create VLANs.
Associate switch ports with VLANs	
awplus(config-vlan)# interface <port-list>	Associate switch ports in access mode with VLANs: Enter Interface Configuration mode for the switch ports that will be in access mode for a particular VLAN. Associate the VLAN with these ports in access mode. Repeat for other VLANs and ports in access mode.
awplus(config-if)# switchport access vlan <vlan-id>	
awplus(config-if)# interface <port-list>	Associate switch ports in trunk mode with VLANs. Enter Interface Configuration mode for all the switch ports that will be in trunk mode for a particular set of VLANs. Set these switch ports to trunk mode. Allow these switch ports to trunk this set of VLANs.
awplus(config-if)# switchport mode trunk [ingress-filter {enable|disable}]	
awplus(config-if)# switchport trunk allowed vlan all or switchport trunk allowed vlan add <vid-list>	
awplus(config-if)# switchport trunk native vlan {<vid>|none}	By default, a trunk mode switch port's native VLAN, the VLAN that the port uses for untagged packet, is VLAN **1**. If required, change the native VLAN from the default.
awplus(config-if)# exit	Return to Global Configuration mode.
awplus(config)# exit	Return to Privileged Exec mode.
awplus# show vlan {all|brief|dynamic|static|<1-4094>}	Confirm VLAN configuration.

Set the Maximum Receive Unit (MRU)

Adding the S-Tag can result in frame sizes that exceed the maximum of 1522 bytes. In order to cope with these larger than normal frames, you should increase the MRU size set for ports configured for double-tagged VLANs. Set the MRU size to:

- 9710 bytes for ports that work at speeds of either 10Mbps or 100Mbs
- 10240 bytes for ports that work at speeds of 1000Mbps

For more information, see the mru command on page 12.5.

Private VLANs

Private VLANs combine the network advantages of conventional VLANs, with an added degree of privacy obtained by limiting the connectivity between selected ports.

This section provides an introduction to:

- Private VLANs for ports in access mode
- Private VLANs for trunked ports

Private VLANs for ports in access mode

An example application of a private VLAN would be a library in which user booths each have a PC with Internet access. In this situation it would usually be undesirable to allow communication between these individual PCs. Connecting the PC to ports within a private isolated VLAN would enable each PC to access the Internet or a library server via a single connection, whilst preventing access between the PCs in the booths.

Another application might be to use private VLANs to simplify IP address assignment. Ports can be isolated from each other whilst still belonging to the same subnet.

A private VLAN comprises the following components:

- a single promiscuous port
- one or more host ports
  There are two types of host ports:
  - **isolated ports**
    These can only communicate with the promiscuous port that is associated with the isolated VLAN.
  - **community ports**
    These can communicate with their associated promiscuous port and other community ports within the community VLAN.
- a single primary VLAN
- one or more secondary VLANS
  There are two types of secondary VLANs:
  - **isolated VLANs**
    In this VLAN type, communication can only take place between each host port and its associated promiscuous port.
  - **community VLANs**
    In this VLAN type, communication can take place between host ports and between each host port and its associated promiscuous port.

Membership rules for private VLANs in access mode

The following membership rules apply when creating and operating private VLANs in access mode.

Each private VLAN:

- must contain one promiscuous port (or aggregated link)
- may contain multiple host ports
- can be configured to span switch instances
- can only contain promiscuous and host ports
- cannot use the default VLAN (vlan1)
- a private *isolated* VLAN can only contain a single promiscuous port
- a private *community* VLAN can contain more than one promiscuous port

A promiscuous port:

- is a member of the primary VLAN and all its associated secondary VLANs
- cannot be a member of both private and non-private VLANs

A host port:

- can be a member of multiple private (community) VLANs, but all these VLANs must share the same promiscuous port
- cannot be a host port in some VLANs and a non-host port in others
- cannot be a promiscuous port in another VLAN

Promiscuous ports

A promiscuous port can communicate with all ports that are members of its associated secondary VLANs. Multiple promiscuous ports can exist in a primary VLAN, but only if the primary VLAN is only associated with community VLANS (that is, that there are no isolated VLANs associated with this port).

A promiscuous port is a member of the primary VLAN and all associated secondary VLANs. Its Port VID is set to the VLAN ID of the primary VLAN.

Host ports

Host ports have two levels of connectivity depending on whether they exist in an isolated or a community VLAN.

1. Host ports within an isolated VLAN

These ports are only allowed to communicate with their VLAN's promiscuous port, even though they share their secondary (isolated) VLAN with other hosts. The host ports receive their data from the promiscuous port via the primary VLAN, and individually transmit their data to the promiscuous port via their common secondary VLAN.

2. Host ports within a community VLAN

These ports are able to communicate with both the promiscuous port and the other ports within the community VLAN that they are associated with. They receive their data from the promiscuous port via the primary VLAN, and transmit their data to both the promiscuous port and the other host ports (within their community VLAN) via their common secondary VLAN. However, the only external path from a community VLAN is from its promiscuous port.

Private VLAN operation with ports in access mode

A basic private VLAN operation is shown in. It comprises primary VLAN 20 plus three secondary VLANS, two community VLANs 21 and 22, and an isolated VLAN 23.

Private VLANs operate within a single switch and comprise one primary VLAN plus a number of secondary VLANS. All data enters the private VLAN ports untagged. Using the example of, data enters the switch via the promiscuous and is forwarded to the host ports using VLAN 20, the primary VLAN. Data returning from the host ports to the promiscuous port (and exiting the switch) use the secondary VLAN associated with its particular host port, VLAN 21, 22, or 23 in the example. Thus the data flows into the switch via the primary VLAN and out of the switch via the secondary VLANs. This situation is not detected outside of the switch, because all its private ports are untagged. Note however, that data flowing between ports within the same community VLAN will do so using the VID of the community VLAN.

Portfast on private VLANS

Within private VLANs, we recommend that you place all host ports into spanning-tree portfast mode and enable BPDU guard. Portfast assumes that because host ports will also be edge ports, they will have no alternative paths (loops) via other bridges. These ports are therefore allowed to move directly from the spanning-tree blocking state into the forwarding state, thus bypassing the intermediate states.

Applying BPDU guard is an extra precaution. This feature disables an edge port if it receives a BPDU frame, because receiving such a frame would indicate that the port has a connection to another network bridge.

For more information on BPDU guard and portfast, see their following commands:

- spanning-tree portfast bpdu-filter command on page 19.58
- spanning-tree portfast (STP) command on page 19.57

Access mode private VLAN configuration example

Table 16-2: Configuration procedure for access mode private VLANs

Command	Description
Create the VLANs	
awplus# `configure terminal`	Enter Global Configuration mode.
awplus(config)# `vlan database`	Enter VLAN Configuration mode.
awplus(config-vlan)# `vlan 20-23`	Create the VLANs.
Create the private VLANs and set the type	
awplus(config-vlan)# `private-vlan 20 primary`	Create primary VLAN 20.
awplus(config-vlan)# `private-vlan 21 community`	Create community VLAN 21.
awplus(config-vlan)# `private-vlan 22 community`	Create community VLAN 22.

Table 16-2: Configuration procedure for access mode private VLANs(cont.)

`awplus(config-vlan)#`
`private-vlan 23 isolated`
	Create isolated VLAN 23.

Associate the secondary VLANs with the primary VLAN	

`awplus(config-vlan)#`
`private-vlan 20 association add 21`
	Associate secondary VLAN 21 with the primary VLAN 20.

`awplus(config-vlan)#`
`private-vlan 20 association add 22`
	Associate secondary VLAN 22 with the primary VLAN 20.

`awplus(config-vlan)#`
`private-vlan 20 association add 23`
	Associate secondary VLAN 23 with the primary VLAN 20.

Set port 1.0.1 to be the promiscuous port	

`awplus(config-if)#`
`exit`
	Return to Global Configuration mode.

`awplus(config)#`
`interface port1.0.1`
	Enter Interface Configuration mode for `port1.0.1`.

`awplus(config-if)#`
`switchport mode private-vlan promiscuous`
	Set the port as a promiscuous ports.

Set the other ports to be host ports	

`awplus(config-if)#`
`exit`
	Return to Global Configuration mode.

`awplus(config)#`
`interface port1.0.2-1.0.4, port1.0.6-1.0.8, port1.0.10-1.0.12`
	Enter Interface Configuration mode for the ports.

`awplus(config-if)#`
`switchport mode private-vlan host`
	Set the ports as host ports.

On the promiscuous port, map the primary VLAN to each of the secondary VLANs	

`awplus(config-if)#`
`exit`
	Return to Global Configuration mode.

`awplus(config)#`
`interface port1.0.1`
	Enter Interface Configuration mode for `port1.0.1`.

`awplus(config-if)#`
`switchport private-vlan mapping 20 add 21-23`
	Associate primary VLAN 20 and the secondary VLANs 21 to 23 to the promiscuous port.

Table 16-2: Configuration procedure for access mode private VLANs(cont.)

Associate the community host ports with the community VLANs	
awplus(config-if)# exit	Return to Global Configuration mode.
awplus(config)# interface port1.0.2-1.0.4	Enter Interface Configuration mode for ports 1.0.2 to 1.0.4.
awplus(config-if)# switchport private-vlan host-association 20 add 21	Associate primary VLAN 20 and secondary VLAN 21 to the host ports.
awplus(config-if)# exit	Return to Global Configuration mode.
awplus(config)# interface port1.0.10-1.0.12	Enter Interface Configuration mode for ports 1.0.10 to 1.0.12.
awplus(config-if)# switchport private-vlan host-association 20 add 22	Associate primary VLAN 20 and secondary VLAN 22 to the host ports.
Associate the isolated host ports with the isolated VLAN 23	
awplus(config-if)# exit	Return to Global Configuration mode.
awplus(config)# interface port1.0.6-1.0.8	Enter Interface Configuration mode for ports 1.0.6 to 1.0.8.
awplus(config-if)# switchport private-vlan host-association 20 add 23	Associate primary VLAN 20 and secondary VLAN 23 to the host ports.

Private VLANs for trunked ports

Private VLAN trunk ports allow you to combine traffic for private isolated VLANs over a trunk. A port in trunk mode enabled as a promiscuous port with the **switchport mode private-vlan trunk promiscuous** command can carry both multiple isolated private VLANs and non-private VLANs. A promiscuous port in trunk mode allows you to combine multiple isolated VLANs on a single trunk port. A port in trunk mode enabled as a secondary port with the **switchport mode private-vlan trunk secondary** command can combine traffic for multiple isolated VLANs over a trunk.

Note Private VLAN trunk ports are not supported by the current AlliedWare Plus GVRP implementation. Private VLAN trunk ports and GVRP are mutually exclusive.

A private VLAN group for trunked ports comprises the following components:

- **a single promiscuous port**
- **one or more isolated secondary ports**
  These can only communicate with the associated promiscuous port.
- **isolated VLANs**
  In this VLAN type, communication can only take place between each secondary port and its associated promiscuous port. Membership rules for private VLANs for trunked ports

The following membership rules apply when creating and operating private VLANs for trunked ports.

A promiscuous trunk port:

- must be in trunk mode
- can be a member of both isolated VLANs and non-isolated VLANs
- has a group ID that is solely used to associate the promiscuous port with secondary ports

A secondary trunk port:

- must be in trunk mode
- can only be a member of isolated VLANs
- cannot be a promiscuous port in another VLAN
- has a group ID that is solely used to associate the secondary port with its promiscuous port

Unlike private VLANs for ports in access mode, private VLANs for trunked ports have no secondary to primary VLAN mappings.

Trunked port private VLAN configuration example

A basic trunked port private VLAN operation is shown in.

The configuration procedure in Table 16-3 show the steps to configure **Switch A**.

Table 16-3: Configuration procedure for Switch A

Command	Description
Create the VLANs	

`awplus#`
`configure terminal`
	Enter Global Configuration mode.

`awplus(config)#`
`vlan database`
	Enter VLAN Configuration mode.

`awplus(config-vlan)#`
`vlan 10,20,30`
	Create the VLANs.

Create the private VLANs and set the type	

`awplus(config-vlan)#`
`private-vlan 10 isolated`
	Create isolated VLAN 10.

`awplus(config-vlan)#`
`private-vlan 20 isolated`
	Create isolated VLAN 20.

`awplus(config-vlan)#`
`private-vlan 30 isolated`
	Create isolated VLAN 30.

Set port 1.0.1 to trunk mode and add the VLANs to be trunked over the port	

`awplus(config-vlan)#`
`interface port1.0.1`
	Enter Interface Configuration mode for `port1.0.1`.

`awplus(config-if)#`
`switchport mode trunk`
	Set the switching characteristics of the port to trunk.

`awplus(config-if)#`
`switchport trunk allowed vlan add 10,20,30`
	Add the VLANs to be trunked over this port.

Set port 1.0.2 to trunk mode and add the VLANs to be trunked over the port	

`awplus(config-if)#`
`exit`
	Return to Global Configuration mode.

`awplus(config)#`
`interface port1.0.2`
	Enter Interface Configuration mode for `port1.0.2`.

`awplus(config-if)#`
`switchport mode trunk`
	Set the switching characteristics of the port to trunk.

`awplus(config-if)#`
`switchport trunk allowed vlan add 10,20`
	Add the VLANs to be trunked over this port.

Table 16-3: Configuration procedure for Switch A(cont.)

Set port 1.0.3 to trunk mode and add the VLANs to be trunked over the port	

`awplus(config-if)#`
`exit`
	Return to Global Configuration mode.

`awplus(config)#`
`interface port1.0.3`
	Enter Interface Configuration mode for port 1.0.3.

`awplus(config-if)#`
`switchport mode trunk`
	Set the switching characteristics of the port to trunk.

`awplus(config-if)#`
`switchport trunk allowed vlan add 10,20,30`
	Add the VLANs to be trunked over this port.

Set port 1.0.1 to be the promiscuous port	

`awplus(config-if)#`
`exit`
	Return to Global Configuration mode.

`awplus(config)#`
`interface port1.0.1`
	Enter Interface Configuration mode for port 1.0.1.

`awplus(config-if)#`
`switchport mode private-vlan trunk promiscuous group 1`
	Enable the port in trunk mode to be promiscuous port for isolated VLANs 10, 20 and 30 with a group ID of 1.

Set port 1.0.2 to be a secondary port	

`awplus(config-if)#`
`exit`
	Return to Global Configuration mode.

`awplus(config)#`
`interface port1.0.2`
	Enter Interface Configuration mode for `port1.0.2`.

`awplus(config-if)#`
`switchport mode private-vlan trunk secondary group 1`
	Enable the port in trunk mode to be a secondary port for isolated VLANs 10 and 20 with a group ID of 1.

Set port 1.0.3 to be a secondary port	

`awplus(config-if)#`
`exit`
	Return to Global Configuration mode.

`awplus(config)#`
`interface port1.0.3`
	Enter Interface Configuration mode for `port1.0.3`.

`awplus(config-if)#`
`switchport mode private-vlan trunk secondary group 1`
	Enable the port in trunk mode to be a secondary port for isolated VLANs 10, 20 and 30 with a group ID of 1.

Chapter 17: VLAN Commands

Command List

This chapter provides an alphabetical reference of commands used to configure VLANs. For more information see Chapter 16, VLAN Introduction.

clear vlan statistics

This command resets the counters for either a specific VLAN statistics instance or (by not specifying an instance) resets the counters for all instances.

Syntax

```
clear vlan statistics [name <instance_name>]
```

Parameter	Description
`vlan statistics`	The count of incoming frames or bytes collected on a per VLAN basis.[1]
`<instance-name>`	The name of the instance for which incoming frames and their bytes are counted.[1]

[1] The terms frame and packet are used interchangeably.

Mode Privileged Exec

Examples To reset all packet counters for the packet counter instance **vlan2-data:**

```
awplus# clear vlan statistics name vlan2-data
```

To reset all packet counters for all packet counter instances.

```
awplus# clear vlan statistics
```

Related Commands show vlan statistics
vlan statistics

private-vlan

Use this command to a create a private VLAN. Private VLANs can be either primary or secondary. Secondary VLANs can be ether community or isolated.

Use the **no** variant of this command to remove the specified private VLAN.

For more information, see the section “Private VLANs” on page 16.6.

Syntax

```
private-vlan <vlan-id> {community|isolated|primary}

no private-vlan <vlan-id> {community|isolated|primary}
```

Parameter	Description
`<vlan-id>`	VLAN ID in the range <2-4094> for the VLAN which is to be made a private VLAN.
`community`	Community VLAN.
`isolated`	Isolated VLAN.
`primary`	Primary VLAN.

Mode VLAN Configuration

Examples

```
            awplus# configure terminal
    awplus(config)# vlan database
awplus(config-vlan)# vlan 2 name vlan2 state enable
awplus(config-vlan)# vlan 3 name vlan3 state enable
awplus(config-vlan)# vlan 4 name vlan4 state enable
awplus(config-vlan)# private-vlan 2 primary
awplus(config-vlan)# private-vlan 3 isolated
awplus(config-vlan)# private-vlan 4 community
```

```
            awplus# configure terminal
    awplus(config)# vlan database
awplus(config-vlan)# no private-vlan 2 primary
awplus(config-vlan)# no private-vlan 3 isolated
awplus(config-vlan)# no private-vlan 4 community
```

private-vlan association

Use this command to associate a secondary VLAN to a primary VLAN. Only one isolated VLAN can be associated to a primary VLAN. Multiple community VLANs can be associated to a primary VLAN.

Use the **no** variant of this command to remove association of all the secondary VLANs to a primary VLAN.

For more information, see the section "Private VLANs" on page 16.6.

Syntax

```
private-vlan <primary-vlan-id> association
    {add <secondary-vlan-id> | remove <secondary-vlan-id>}

no private-vlan <primary-vlan-id> association
```

Parameter	Description
<primary-vlan-id>	VLAN ID of the primary VLAN.
<secondary-vlan-id>	VLAN ID of the secondary VLAN (either isolated or community).

Mode VLAN Configuration

Examples

```
           awplus# configure terminal
   awplus(config)# vlan database
awplus(config-vlan)# private-vlan 2 association add 3
awplus(config-vlan)# private-vlan 2 association remove 3
awplus(config-vlan)# no private-vlan 2 association
```

show vlan

Use this command to display information about a particular VLAN by specifying the VLAN ID. It displays information for all the VLANs configured.

Syntax `show vlan {all|brief|dynamic|static|<1-4094>}`

Parameter	Description
<1-4094>	Display information about the VLAN specified by the VLAN ID.
`all`	Display information about all VLANs on the device.
`brief`	Display information about all VLANs on the device.
`dynamic`	Display information about all VLANs learned dynamically.
`static`	Display information about all statically configured VLANs.

Mode User Exec and Privileged Exec

Example To display information about VLAN 2, use the command:

```
awplus# show vlan 2
```

Output Figure 17-1: Example output from the **show vlan** command

```
VLAN ID  Name             Type    State   Member ports
                                          (u)-Untagged, (t)-Tagged
======= ================ ======= ======= ====================================
2       VLAN0002         STATIC  ACTIVE  port1.0.5(u) port1.0.6(u) port1.0.7(u)
                                          port1.0.8(u)
.
.
```

Related Commands vlan

show vlan classifier group

Use this command to display information about all configured VLAN classifier groups or a specific group.

Syntax

```
show vlan classifier group [<1-16>]
```

Parameter	Description
<1-16>	VLAN classifier group identifier

Mode User Exec and Privileged Exec

Usage If a group ID is not specified, all configured VLAN classifier groups are shown. If a group ID is specified, a specific configured VLAN classifier group is shown.

Example To display information about VLAN classifier group 1, enter the command:

```
awplus# show vlan classifier group 1
```

Related Commands vlan classifier group

show vlan classifier group interface

Use this command to display information about a single switch port interface for all configured VLAN classifier groups.

Syntax

```
show vlan classifier group interface <switch-port>
```

Parameter	Description
<switch-port>	Specify the switch port interface classifier group identifier

Mode User Exec and Privileged Exec

Usage All configured VLAN classifier groups are shown for a single interface.

Example To display VLAN classifier group information for switch port interface `port1.0.2`, enter the command:

```
awplus# show vlan classifier group interface port1.0.2
```

Output To display VLAN classifier group information for switch port interface `port1.0.2`, enter the command:

```
awplus# show vlan classifier group interface port1.0.2
```

Related Commands vlan classifier group
show vlan classifier interface group

show vlan classifier interface group

Use this command to display information about all interfaces configured for a VLAN group or all the groups.

Syntax

```
show vlan classifier interface group [<1-16>]
```

Parameter	Description
<1-16>	VLAN classifier interface group identifier

Mode User Exec and Privileged Exec

Usage If a group ID is not specified, all interfaces configured for all VLAN classifier groups are shown. If a group ID is specified, the interfaces configured for this VLAN classifier group are shown.

Example To display information about all interfaces configured for all VLAN groups, enter the command:

```
awplus# show vlan classifier interface group
```

To display information about all interfaces configured for VLAN group 1, enter the command:

```
awplus# show vlan classifier interface group 1
```

Output Figure 17-2: Example output from the **show vlan classifier interface group** command

```
vlan classifier group 1 interface port1.0.1
vlan classifier group 1 interface port1.0.2
vlan classifier group 2 interface port1.0.3
vlan classifier group 2 interface port1.0.4
```

Output Figure 17-3: Example output from the **show vlan classifier interface group 1** command

```
vlan classifier group 1 interface port1.0.1
vlan classifier group 1 interface port1.0.2
```

Related Commands vlan classifier group
show vlan classifier group interface

show vlan classifier rule

Use this command to display information about all configured VLAN classifier rules or a specific rule.

Syntax

```
show vlan classifier rule [<1-256>]
```

Parameter	Description
<1-256>	VLAN classifier rule identifier

Mode User Exec and Privileged Exec

Usage If a rule ID is not specified, all configured VLAN classifier rules are shown. If a rule ID is specified, a specific configured VLAN classifier rule is shown.

Example To display information about VLAN classifier rule 1, enter the command:

```
awplus# show vlan classifier rule 1
```

Output Figure 17-4: Example output from the **show vlan classifier rule 1** command

```
vlan classifier group 1 add rule 1
```

Related Commands

vlan classifier activate
vlan classifier rule ipv4
vlan classifier rule proto

show vlan private-vlan

Use this command to display the private VLAN configuration and associations.

Syntax `show vlan private-vlan`

Mode User Exec and Privileged Exec

Example To display the private VLAN configuration and associations, enter the command:

```
awplus# show vlan private-vlan
```

Output Figure 17-5: Example output from the **show vlan private-vlan** command

```
awplus#show vlan private-vlan
 PRIMARY       SECONDARY          TYPE         INTERFACES
 -------       ---------        ----------     ----------
       2               3          isolated
       2               4         community
                       8          isolated
```

Related Commands private-vlan
private-vlan association

show vlan statistics

Use this command to display the current configuration for either a specific VLAN statistics instance, or (by not specifying an instance) display all VLAN packet counter instances.

Syntax

```
show vlan statistics [name <instance_name>]
```

Mode

User Exec and Privileged Exec

Examples

To display all packet counters for the packet counter instance **vlan2-data**

```
awplus# show vlan statistics name vlan2-data
```

To display all packet counters for all packet counter instances.

```
awplus# show vlan statistics
```

Figure 17-6: Example output from the **show vlan statistics** command

```
VLAN Stats Collection: vlan2-data
  VLAN ID: 2
  Port Map: port1.0.1, port1.0.2, port1.0.4
  Ingress Packets: total 941, bytes 66185
```

Related Commands

clear vlan statistics
vlan statistics

switchport access vlan

Use this command to change the port-based VLAN of the current port.

Use the **no** variant of this command to change the port-based VLAN of this port to the default VLAN, *vlan1*.

Syntax

```
switchport access vlan <vlan-id>
```

```
no switchport access vlan
```

Parameter	Description
<vlan-id>	<1-4094> The port-based VLAN ID for the port.

Default Reset the default VLAN 1 to specified switchports using the negated form of this command.

Mode Interface Configuration

Usage Any untagged frame received on this port will be associated with the specified VLAN.

Examples To change the port-based VLAN to VLAN 3 for `port1.0.2`, use the commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.2
awplus(config-if)# switchport access vlan 3
```

To reset the port-based VLAN to the default VLAN 1 for `port1.0.2`, use the commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.2
awplus(config-if)# no switchport access vlan
```

Validation Command show interface switchport

Related Commands show vlan

switchport enable vlan

This command enables the VLAN on the port manually once disabled by certain actions, such as QSP (QoS Storm Protection) or EPSR (Ethernet Protection Switching Ring). Note that if the VID is not given, all disabled VLANs are re-enabled.

Syntax

```
switchport enable vlan [<1-4094>]
```

Parameter	Description
`vlan`	Re-enables the VLAN on the port.
<1-4094>	VLAN ID.

Mode Interface Configuration

Example To re-enable the `port1.0.1` from VLAN 1:

```
        awplus# configure terminal
awplus(config)# interface port1.0.1
awplus(config-if)# switchport enable vlan 1
```

Related Commands show mls qos interface storm-status
storm-window

switchport mode access

Use this command to set the switching characteristics of the port to access mode. Received frames are classified based on the VLAN characteristics, then accepted or discarded based on the specified filtering criteria.

Syntax

```
switchport mode access [ingress-filter {enable|disable}]
```

Parameter	Description
`ingress-filter`	Set the ingress filtering for the received frames.
`enable`	Turn on ingress filtering for received frames. This is the default.
`disable`	Turn off ingress filtering to accept frames that do not meet the classification criteria.

Default By default, ports are in access mode with ingress filtering on.

Usage Use access mode to send untagged frames only.

Mode Interface Configuration

Example

```
         awplus# configure terminal
 awplus(config)# interface port1.0.2
awplus(config-if)# switchport mode access ingress-filter
                   enable
```

Validation Command show interface switchport

switchport mode private-vlan

Use this command to make a Layer 2 port a private VLAN host port or a promiscuous port.

Use the **no** variant of this command to remove the configuration.

Syntax

```
switchport mode private-vlan {host|promiscuous}
```

```
no switchport mode private-vlan {host|promiscuous}
```

Parameter	Description
`host`	This port type can communicate with all other host ports assigned to the same community VLAN, but it cannot communicate with the ports in the same isolated VLAN. All communications outside of this VLAN must pass through a promiscuous port in the associated primary VLAN.
`promiscuous`	A promiscuous port can communicate with all interfaces, including the community and isolated ports within a private VLAN.

Mode Interface Configuration

Examples

```
          awplus#  configure terminal
  awplus(config)#  interface port1.0.2
awplus(config-if)#  switchport mode private-vlan host
  awplus(config)#  interface port1.0.3
awplus(config-if)#  switchport mode private-vlan promiscuous
  awplus(config)#  interface port1.0.4
awplus(config-if)#  no switchport mode private-vlan promiscuous
```

Related Commands switchport private-vlan mapping

switchport mode private-vlan trunk secondary

Use this command to enable a port in trunk mode to be a secondary port for isolated VLANs.

Note Private VLAN trunk ports are not supported by the current AlliedWare Plus GVRP implementation. Private VLAN trunk ports and GVRP are mutually exclusive.

Use the **no** variant of this command to remove a port in trunk mode as a secondary port for isolated VLANs.

Syntax

```
switchport mode private-vlan trunk secondary group <group-id>

no switchport mode private-vlan trunk secondary
```

Parameter	Description
<group-id>	The group ID is a numeric value in the range 1 to 32 that is used to associate a secondary port with its promiscuous port.

Default By default, a port in trunk mode is disabled as a secondary port.

When a port in trunk mode is enabled to be a secondary port for isolated VLANs, by default it will have a native VLAN of **none** (no native VLAN specified).

Mode Interface Configuration

Usage A port must be put in trunk mode with switchport mode trunk command before the port is enabled as a secondary port in trunk mode.

To add VLANs to be trunked over the secondary port use the switchport trunk allowed vlan command. These must be isolated VLANs and must exist on the associated promiscuous port.

To configure the native VLAN for the secondary port, use the switchport trunk native vlan command. The native VLAN must be an isolated VLAN and must exist on the associated promiscuous port.

For further information, see "Private VLANs for trunked ports" on page 16.11.

Examples To create isolated private VLAN 2 and then enable `port1.0.3` in trunk mode as a secondary port for the this VLAN with the group ID of 3, use the following commands:

```
           awplus#  configure terminal
   awplus(config)#  vlan database
awplus(config-vlan)#  vlan 2
awplus(config-vlan)#  private-vlan 2 isolated
awplus(config-vlan)#  exit
   awplus(config)#  interface port1.0.3
 awplus(config-if)#  switchport mode trunk
 awplus(config-if)#  switchport trunk allowed vlan add 2
 awplus(config-if)#  switchport mode private-vlan trunk
                     secondary group 3
```

```
         awplus#  configure terminal
 awplus(config)#  interface port1.0.3
awplus(config-if)#  no switchport mode private-vlan trunk
                    secondary
```

Related Commands

switchport mode private-vlan trunk promiscuous
switchport mode trunk
switchport trunk allowed vlan
switchport trunk native vlan
show vlan private-vlan

switchport mode private-vlan trunk promiscuous

Use this command to enable a port in trunk mode to be promiscuous port for isolated VLANs.

Note Private VLAN trunk ports are not supported by the current AlliedWare Plus GVRP implementation. Private VLAN trunk ports and GVRP are mutually exclusive.

Use the **no** variant of this command to remove a port in trunk mode as a promiscuous port for isolated VLANs. You must first remove the secondary port, or ports, in trunk mode associated with the promiscuous port with the **no switchport mode private-vlan trunk secondary** command.

Syntax

```
switchport mode private-vlan trunk promiscuous group <group-id>

no switchport mode private-vlan trunk promiscuous
```

Parameter	Description
<group-id>	The group ID is a numeric value in the range 1 to 32 that is used to associate the promiscuous port with secondary ports.

Default By default, a port in trunk mode is disabled as a promiscuous port.

Mode Interface Configuration

Usage A port must be put in trunk mode with switchport mode trunk command before it can be enabled as a promiscuous port.

To add VLANs to be trunked over the promiscuous port, use the switchport trunk allowed vlan command. These VLANs can be isolated VLANs, or non-private VLANs.

To configure the native VLAN for the promiscuous port, use the switchport trunk native vlan command. The native VLAN can be an isolated VLAN, or a non-private VLAN.

When you enable a promiscuous port, all of the secondary port VLANs associated with the promiscuous port via the group ID number must be added to the promiscuous port. In other words, the set of VLANs on the promiscuous port must be a superset of all the VLANs on the secondary ports within the group.

For further information, see "Private VLANs for trunked ports" on page 16.11.

Examples To create the isolated VLANs 2, 3 and 4 and then enable `port1.0.2` in trunk mode as a promiscuous port for these VLANs with the group ID of 3, use the following commands:

```
           awplus# configure terminal
   awplus(config)# vlan database
awplus(config-vlan)# vlan 2-4
awplus(config-vlan)# private-vlan 2 isolated
awplus(config-vlan)# private-vlan 3 isolated
awplus(config-vlan)# private-vlan 4 isolated
awplus(config-vlan)# exit
     awplus(config)# interface port1.0.2
  awplus(config-if)# switchport mode trunk
  awplus(config-if)# switchport trunk allowed vlan add 2-4
  awplus(config-if)# switchport mode private-vlan trunk
                     promiscuous group 3
```

To remove port1.0.2 in trunk mode as a promiscuous port for a private VLAN, use the commands:

```
           awplus# configure terminal
   awplus(config)# interface port1.0.2
awplus(config-if)# no switchport mode private-vlan trunk
                   promiscuous
```

Note that you must remove the secondary port or ports enabled as trunk ports that are associated with the promiscuous port before removing the promiscuous port.

Related Commands

switchport mode private-vlan trunk secondary
switchport mode trunk
switchport trunk allowed vlan
switchport trunk native vlan
show vlan private-vlan

switchport mode trunk

Use this command to set the switching characteristics of the port to trunk. Received frames are classified based on the VLAN characteristics, then accepted or discarded based on the specified filtering criteria.

Syntax

```
switchport mode trunk [ingress-filter {enable|disable}]
```

Parameter	Description
`ingress-filter`	Set the ingress filtering for the frames received.
`enable`	Turn on ingress filtering for received frames. This is the default.
`disable`	Turn off ingress filtering to accept frames that do not meet the classification criteria.

Default By default, ports are in access mode, are untagged members of the default VLAN (vlan1), and have ingress filtering on.

Mode Interface Configuration

Usage A port in trunk mode can be a tagged member of multiple VLANs, and an untagged member of one native VLAN.

To configure which VLANs this port will trunk for, use the switchport trunk allowed vlan command.

Example

```
          awplus# configure terminal
  awplus(config)# interface port1.0.3
awplus(config-if)# switchport mode trunk ingress-filter enable
```

Validation Command show interface switchport

switchport private-vlan host-association

Use this command to associate a primary VLAN and a secondary VLAN to a host port. Only one primary and secondary VLAN can be associated to a host port.

Use the **no** variant of this command to remove the association.

Syntax

```
switchport private-vlan host-association <primary-vlan-id> add
    <secondary-vlan-id>

no switchport private-vlan host-association
```

Parameter	Description
<primary-vlan-id>	VLAN ID of the primary VLAN.
<secondary-vlan-id>	VLAN ID of the secondary VLAN (either isolated or community).

Mode Interface Configuration

Examples

```
          awplus# configure terminal
  awplus(config)# interface port1.0.2
awplus(config-if)# switchport private-vlan host-association 2
                   add 3
```

```
          awplus# configure terminal
  awplus(config)# interface port1.0.2
awplus(config-if)# no switchport private-vlan host-association
```

switchport private-vlan mapping

Use this command to associate a primary VLAN and a set of secondary VLANs to a promiscuous port.

Use the **no** variant of this to remove all the association of secondary VLANs to primary VLANs for a promiscuous port.

Syntax

```
switchport private-vlan mapping <primary-vlan-id> add
   <secondary-vid-list>

switchport private-vlan mapping <primary-vlan-id> remove
   <secondary-vid-list>

no switchport private-vlan mapping
```

Parameter	Description
<primary-vlan-id>	VLAN ID of the primary VLAN.
<secondary-vid-list>	VLAN ID of the secondary VLAN (either isolated or community), or a range of VLANs, or a comma-separated list of VLANs and ranges.

Mode Interface Configuration

Usage This command can be applied to a switch port or a static channel group, but not a dynamic (LACP) channel group. LACP channel groups (dynamic/LACP aggregators) cannot be promiscuous ports in private VLANs.

Examples

```
          awplus#  configure terminal
  awplus(config)#  interface port1.0.2
awplus(config-if)# switchport private-vlan mapping 2 add 3-4
awplus(config-if)# switchport private-vlan mapping 2 remove 3-4
awplus(config-if)# no switchport private-vlan mapping
```

Related Commands switchport mode private-vlan

switchport trunk allowed vlan

Use this command to add VLANs to be trunked over this switch port. Traffic for these VLANs can be sent and received on the port.

Use the **no** variant of this command to reset switching characteristics of a specified interface to negate a trunked configuration specified with **switchport trunk allowed vlan** command.

Syntax

```
switchport trunk allowed vlan all

switchport trunk allowed vlan none

switchport trunk allowed vlan add <vid-list>

switchport trunk allowed vlan remove <vid-list>

switchport trunk allowed vlan except <vid-list>

no switchport trunk
```

Parameter	Description
`all`	Allow all VLANs to transmit and receive through the port.
`none`	Allow no VLANs to transmit and receive through the port.
`add`	Add a VLAN to transmit and receive through the port. Only use this parameter if a list of VLANs are already configured on a port.
`remove`	Remove a VLAN from transmit and receive through the port. Only use this parameter if a list of VLANs are already configured on a port.
`except`	All VLANs, except the VLAN for which the VID is specified, are part of its port member set. Only use this parameter to remove VLANs after either this parameter or the **all** parameter have added VLANs to a port.
`<vid-list>`	<2-4094> The ID of the VLAN or VLANs that will be added to, or removed from, the port. A single VLAN, VLAN range, or comma-separated VLAN list can be set. For a VLAN range, specify two VLAN numbers: lowest, then highest number in the range, separated by a hyphen. For a VLAN list, specify the VLAN numbers separated by commas. Do not enter spaces between hyphens or commas when setting parameters for VLAN ranges or lists.

Default By default, ports are untagged members of the default VLAN (vlan1).

Mode Interface Configuration

Usage The **all** parameter sets the port to be a tagged member of all the VLANs configured on the device. The **none** parameter removes all VLANs from the port's tagged member set. The **add** and **remove** parameters will add and remove VLANs to and from the port's member set. See the note below about restrictions when using the **add**, **remove**, **except**, and **all** parameters.

Note: Only use the **add** or the **remove** parameters with this command if a list of VLANs are configured on a port. Only use the **except** parameter to remove VLANs after either the **except** or the **all** parameters have first been used to add a list of VLANs to a port.

To remove a VLAN, where the configuration for `port1.0.18` shows the below output:

```
awplus#show running-config
!
interface port1.0.18
switchport
switchport mode trunk
switchport trunk allowed vlan except 4
```

Remove VLAN 3 by re-entering the **except** parameter with the list of VLANs to remove, instead of using the **remove** parameter, as shown in the command example below:

```
         awplus# configure terminal
 awplus(config)# interface port1.0.18
awplus(config-if)# switchport trunk allowed vlan except 3,4
```

Then the configuration is changed after entering the above commands to remove VLAN 3:

```
awplus#show running-config
!
interface port1.0.18
switchport
switchport mode trunk
switchport trunk allowed vlan except 3-4
```

To add a VLAN, where the configuration for `port1.0.18` shows the below output:

```
awplus#show running-config
!
interface port1.0.18
switchport
switchport mode trunk
switchport trunk allowed vlan except 3-5
```

Add VLAN 4 by re-entering the **except** parameter with a list of VLANs to exclude, instead of using the **add** parameter to include VLAN 4, as shown in the command example below:

```
         awplus# configure terminal
 awplus(config)# interface port1.0.18
awplus(config-if)# switchport trunk allowed vlan except 3,5
```

The configuration is changed after entering the above commands to add VLAN 4:

```
awplus#show running-config
!
interface port1.0.18
switchport
switchport mode trunk
switchport trunk allowed vlan except 3,5
```

Examples The following shows adding a single VLAN to the port's member set.

```
         awplus# configure terminal
 awplus(config)# interface port1.0.2
awplus(config-if)# switchport trunk allowed vlan add 2
```

The following shows adding a range of VLANs to the port's member set.

```
         awplus# configure terminal
 awplus(config)# interface port1.0.2
awplus(config-if)# switchport trunk allowed vlan add 2-4
```

The following shows adding a list of VLANs to the port's member set.

```
         awplus# configure terminal
 awplus(config)# interface port1.0.2
awplus(config-if)# switchport trunk allowed vlan add 2,3,4
```

switchport trunk native vlan

Use this command to configure the native VLAN for this port. The native VLAN is used for classifying the incoming untagged packets. Use the **none** parameter with this command to remove the native VLAN from the port and set the acceptable frame types to vlan-tagged only.

Use the **no** variant of this command to revert the native VLAN to the default VLAN ID 1. Command negation removes tagged VLANs, and sets the native VLAN to the default VLAN.

Syntax

```
switchport trunk native vlan {<vid>|none}

no switchport trunk native vlan
```

Parameter	Description
`<vid>`	`<2-4094>` The ID of the VLAN that will be used to classify the incoming untagged packets. The VLAN ID must be a part of the VLAN member set of the port.
`none`	No native VLAN specified. This option removes the native VLAN from the port and sets the acceptable frame types to vlan-tagged only. Note: Use the **no** variant of this command to revert to the default VLAN 1 as the native VLAN for the specified interface switchport - not **none**.

Default VLAN 1 (the default VLAN), which is reverted to using the **no** form of this command.

Mode Interface Configuration

Examples The following commands show configuration of VLAN 2 as the native VLAN for interface `port1.0.2`:

```
          awplus#  configure terminal
  awplus(config)#  interface port1.0.2
awplus(config-if)#  switchport trunk native vlan 2
```

The following commands show the removal of the native VLAN for interface `port1.0.2`:

```
          awplus#  configure terminal
  awplus(config)#  interface port1.0.2
awplus(config-if)#  switchport trunk native vlan none
```

The following commands revert the native VLAN to the default VLAN 1 for interface `port1.0.2`:

```
          awplus#  configure terminal
  awplus(config)#  interface port1.0.2
awplus(config-if)#  no switchport trunk native vlan
```

switchport voice dscp

Use this command to configure the Layer 3 DSCP value advertised when the transmission of LLDP-MED Network Policy TLVs for voice devices is enabled. When LLDP-MED capable IP phones receive this network policy information, they transmit voice data with the specified DSCP value.

Use the **no** variant of this command to reset the DSCP value to the default, 0.

Syntax

```
switchport voice dscp <0-63>
```

```
no switchport voice dscp
```

Parameter	Description
`dscp`	Specify a DSCP value for voice data.
`<0-63>`	DSCP value.

Default A DSCP value of 0 will be advertised.

Mode Interface Configuration

Usage LLDP-MED advertisements including Network Policy TLVs are transmitted via a port if:

- LLDP is enabled (lldp run command on page 66.17)
- Voice VLAN is configured for the port (switchport voice vlan command on page 17.28)
- The port is configured to transmit LLDP advertisements—enabled by default (lldp transmit receive command on page 66.21)
- The port is configured to transmit Network Policy TLVs—enabled by default (lldp med-tlv-select command on page 66.10)
- There is an LLDP-MED device connected to the port

Example

```
          awplus# configure terminal
  awplus(config)# interface port1.0.5
awplus(config-if)# switchport voice dscp 27
```

Related Commands

lldp med-tlv-select
show lldp
switchport voice vlan

switchport voice vlan

Use this command to configure the Voice VLAN tagging advertised when the transmission of LLDP-MED Network Policy TLVs for voice endpoint devices is enabled. When LLDP-MED capable IP phones receive this network policy information, they transmit voice data with the specified tagging. This command also sets the ports to be spanning tree edge ports, that is, it enables spanning tree portfast on the ports.

Use the **no** variant of this command to remove LLDP-MED network policy configuration for voice devices connected to these ports. This does not change the spanning tree edge port status.

Syntax

```
switchport voice vlan [<vid>|dot1p|dynamic|untagged]
```

```
no switchport voice vlan
```

Parameter	Description
`<vid>`	VLAN identifier, in the range 1 to 4094.
`dot1p`	The IP phone should send User Priority tagged packets, that is, packets in which the tag contains a User Priority value, and a VID of 0. (The User Priority tag is also known as the 802.1p priority tag, or the Class of Service (CoS) tag.)
`dynamic`	The VLAN ID with which the IP phone should send tagged packets will be assigned by RADIUS authentication.
`untagged`	The IP phone should send untagged packets.

Default By default, no Voice VLAN is configured, and therefore no network policy is advertised for voice devices.

Mode Interface Configuration

Usage LLDP-MED advertisements including Network Policy TLVs are transmitted via a port if:

- LLDP is enabled (lldp run command on page 66.17)
- Voice VLAN is configured for the port using this command (switchport voice vlan)
- The port is configured to transmit LLDP advertisements—enabled by default (lldp transmit receive command on page 66.21)
- The port is configured to transmit Network Policy TLVs—enabled by default (lldp med-tlv-select command on page 66.10)
- There is an LLDP-MED device connected to the port.

To set the priority value to be advertised for tagged frames, use the switchport voice vlan priority command on page 17.30.

If the Voice VLAN details are to be assigned by RADIUS, then the RADIUS server must be configured to send the attribute 'Egress-VLANID (56)' or 'Egress-VLAN-Name (58)' in the RADIUS Accept message when authenticating a phone attached to this port. To set these attributes on the local RADIUS server, use the egress-vlan-id command on page 48.19 or the egress-vlan-name command on page 48.20.

For more information about configuring authentication for Voice VLAN, "Configuring LLDP" on page 65.11.

If the ports have been set to be edge ports by the switchport voice vlan command, the **no** variant of this command will leave them unchanged as edge ports. To set them back to their default non-edge port configuration, use the spanning-tree edgeport (RSTP and MSTP) command on page 19.38.

Examples To tell IP phones connected to `port1.0.5` to send voice data tagged for VLAN 10, use the commands:

```
        awplus#  configure terminal
    awplus(config)#  interface port1.0.5
 awplus(config-if)#  switchport voice vlan 10
```

To tell IP phones connected to ports 1.0.8-1.0.12 to send priority tagged packets (802.1p priority tagged with VID 0, so that they will be assigned to the port VLAN) use the following commands. The priority value is 5 by default, but can be configured with the switchport voice vlan priority command.

```
        awplus#  configure terminal
    awplus(config)#  interface port1.0.8-port1.0.12
 awplus(config-if)#  switchport voice vlan dot1p
```

To dynamically configure the VLAN ID advertised to IP phones connected to `port1.0.1` based on the VLAN assigned by RADIUS authentication (with RADIUS attribute 'Egress-VLANID' or 'Egress-VLAN-Name' in the RADIUS accept packet), use the commands:

```
        awplus#  configure terminal
    awplus(config)#  interface port1.0.1
 awplus(config-if)#  switchport voice vlan dynamic
```

To remove the Voice VLAN, and therefore disable the transmission of LLDP-MED network policy information for voice devices on `port1.0.24`, use the following commands:

```
        awplus#  configure terminal
    awplus(config)#  interface port1.0.24
 awplus(config-if)#  no switchport voice vlan
```

Related Commands

egress-vlan-id
egress-vlan-name
lldp med-tlv-select
spanning-tree edgeport (RSTP and MSTP)
switchport voice dscp
switchport voice vlan priority
show lldp

switchport voice vlan priority

Use this command to configure the Layer 2 user priority advertised when the transmission of LLDP-MED Network Policy TLVs for voice devices is enabled. This is the priority in the User Priority field of the IEEE 802.1Q VLAN tag, also known as the Class of Service (CoS), or 802.1p priority. When LLDP-MED capable IP phones receive this network policy information, they transmit voice data with the specified priority.

Syntax

```
switchport voice vlan priority <0-7>
```

```
no switchport voice vlan priority
```

Parameter	Description
`priority`	Specify a user priority value for voice data.
`<0-7>`	Priority value.

Default By default, the Voice VLAN user priority value is 5.

Mode Interface Configuration

Usage LLDP-MED advertisements including Network Policy TLVs are transmitted via a port if:

- LLDP is enabled (lldp run command on page 66.17)
- Voice VLAN is configured for the port (switchport voice vlan command on page 17.28)
- The port is configured to transmit LLDP advertisements—enabled by default (lldp transmit receive command on page 66.21)
- The port is configured to transmit Network Policy TLVs—enabled by default (lldp med-tlv-select command on page 66.10)
- There is an LLDP-MED device connected to the port.

To set the Voice VLAN tagging to be advertised, use the switchport voice vlan command on page 17.28.

Example To tell IP phones connected to `port1.0.5` to send voice data with a user priority value of 6, use the commands:

```
        awplus#  configure terminal
awplus(config)#  interface port1.0.5
awplus(config-if)#  switchport voice vlan priority 6
```

Related Commands

lldp med-tlv-select
show lldp
switchport voice vlan

vlan

This command creates VLANs, assigns names to them, and enables or disables them. Specifying the `disable` state causes all forwarding over the specified VLAN ID to cease. Specifying the `enable` state allows forwarding of frames on the specified VLAN.

The **no** variant of this command destroys the specified VLANs.

Syntax

```
vlan <vid> [name <vlan-name>] [state {enable|disable}]

vlan <vid-range> [state {enable|disable}]

vlan {<vid>|<vlan-name>} [mtu <mtu-value>]

no vlan {<vid>|<vid-range>} [mtu]
```

Parameter	Description
`<vid>`	The VID of the VLAN to enable or disable in the range <**1-4094**>.
`<vlan-name>`	The ASCII name of the VLAN. Maximum length: **32** characters.
`<vid-range>`	Specifies a range of VLAN identifiers.
`<mtu-value>`	Specifies the Maximum Transmission Unit (MTU) size in bytes, in the range 68 to1500 bytes, for the VLAN.
`enable`	Sets VLAN into an `enable` state.
`disable`	Sets VLAN into a `disable` state.

Default By default, VLANs are enabled when they are created.

Mode VLAN Configuration

Examples

```
           awplus# configure terminal
   awplus(config)# vlan database
awplus(config-vlan)# vlan 45 name accounts state enable
```

```
           awplus# configure terminal
   awplus(config)# vlan database
awplus(config-vlan)# no vlan 45
```

Related Commands mtu
vlan database
show vlan

vlan classifier activate

Use this command in Interface Configuration mode to associate a VLAN classifier group with the switch port, and optionally associate the group with a VLAN identifier and a switch port.

Use the **no** variant of this command to remove the VLAN classifier group from the switch port, and a VLAN if the VLAN classifier group was also associated with a VLAN identifier.

Syntax

```
vlan classifier activate <vlan-class-group-id> [vlan <VID>]

no vlan classifier activate <vlan-class-group-id>
```

Parameter	Description
<vlan-class-group-id>	Specify a VLAN classifier group identifier in the range <1-16>.
<VID>	Specify a VLAN identifier in the range <1-4094>.

Mode Interface Configuration mode for a switch port.

Example To associate VLAN classifier group 3 with switch port1.0.3, enter the following commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.3
awplus(config-if)# vlan classifier activate 3
```

To associate VLAN classifier group 3 with switch port1.0.3 and the default VLAN 1, enter the following commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.3
awplus(config-if)# vlan classifier activate 3 vlan 1
```

To remove VLAN classifier group 3 from switch port1.0.3, enter the following commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.3
awplus(config-if)# no vlan classifier activate 3
```

Related Commands

show vlan classifier rule
vlan classifier group
vlan classifier rule ipv4
vlan classifier rule proto

vlan classifier group

Use this command to create a group of VLAN classifier rules. The rules must already have been created.

Use the **no** variant of this command to delete a group of VLAN classifier rules.

Syntax

```
vlan classifier group <1-16> {add|delete} rule <vlan-class-rule-id>
```

```
no vlan classifier group <1-16>
```

Parameter	Description
<1-16>	VLAN classifier group identifier
`add`	Add the rule to the group.
`delete`	Delete the rule from the group.
<vlan-class-rule-id>	The VLAN classifier rule identifier.

Mode Global Configuration

Example

```
        awplus# configure terminal
awplus(config)# vlan classifier group 3 add rule 5
```

Related Commands

show vlan classifier rule
vlan classifier activate
vlan classifier rule ipv4
vlan classifier rule proto

vlan classifier rule ipv4

Use this command to create an IPv4 subnet-based VLAN classifier rule and map it to a specific VLAN. Use the **no** variant of this command to delete the VLAN classifier rule.

Syntax

```
vlan classifier rule <1-256> ipv4 <ip-addr/prefix-length>  vlan
    <1-4094>

no vlan classifier rule <1-256>
```

Parameter	Description
<1-256>	Specify the VLAN Classifier Rule identifier.
<ip-addr/prefix-length>	Specify the IP address and prefix length.
<1-4094>	Specify a VLAN ID to which an untagged packet is mapped in the range *<1-4094>*.

Mode Global Configuration

Usage If the source IP address matches the IP subnet specified in the VLAN classifier rule, the received packets are mapped to the specified VLAN.

Example

```
        awplus# configure terminal
awplus(config)# vlan classifier rule 3 ipv4 3.3.3.3/8 vlan 5
```

Related Commands

show vlan classifier rule
vlan classifier activate
vlan classifier rule proto

vlan classifier rule proto

Use this command to create a protocol type-based VLAN classifier rule, and map it to a specific VLAN.

The **no** variant of this command destroys the rule.

Syntax

```
vlan classifier rule <1-256> proto <protocol>
   encap {ethv2|nosnapllc|snapllc} vlan <1-4094>

no vlan classifier rule <1-256>
```

<table>
<tr><th>Parameter</th><th>Description</th></tr>
<tr><td><i>&lt;1-256&gt;</i></td><td>VLAN Classifier identifier</td></tr>
<tr><td>proto</td><td>Protocol type</td></tr>
<tr><td>&lt;protocol&gt;</td><td>Specify a protocol either by its decimal number (0-65535) or by one of the following protocol names:
<table>
<tr><td>[arp|2054]</td><td>Address Resolution protocol</td></tr>
<tr><td>[atalkaarp|33011]</td><td>Appletalk AARP protocol</td></tr>
<tr><td>[atalkddp|32923]</td><td>Appletalk DDP protocol</td></tr>
<tr><td>[atmmulti|34892]</td><td>MultiProtocol Over ATM protocol</td></tr>
<tr><td>[atmtransport|34948]</td><td>Frame-based ATM Transport protocol</td></tr>
<tr><td>[dec|24576]</td><td>DEC Assigned protocol</td></tr>
<tr><td>[deccustom|24582]</td><td>DEC Customer use protocol</td></tr>
<tr><td>[decdiagnostics|24581]</td><td>DEC Systems Comms Arch protocol</td></tr>
<tr><td>[decdnadumpload|24577]</td><td>DEC DNA Dump/Load protocol</td></tr>
<tr><td>[decdnaremoteconsole|24578]</td><td>DEC DNA Remote Console protocol</td></tr>
<tr><td>[decdnarouting|24579]</td><td>DEC DNA Routing protocol</td></tr>
<tr><td>[declat|24580]</td><td>DEC LAT protocol</td></tr>
<tr><td>[decsyscomm|24583]</td><td>DEC Systems Comms Arch protocol</td></tr>
<tr><td>[g8bpqx25|2303]</td><td>G8BPQ AX.25 protocol</td></tr>
<tr><td>[ieeeaddrtrans|2561]</td><td>Xerox IEEE802.3 PUP Address</td></tr>
</table>
</td></tr>
</table>

Parameter(cont.)	Description(cont.)	
	[ieeepup|2560]	Xerox IEEE802.3 PUP protocol
	[ip|2048]	IP protocol
	[ipx|33079]	IPX protocol
	[netbeui|61680]	IBM NETBIOS/NETBEUI protocol
	[netbeui|61681]	IBM NETBIOS/NETBEUI protocol
	[pppdiscovery|34915]	PPPoE discovery protocol
	[pppsession|34916]	PPPoE session protocol
	[rarp|32821]	Reverse Address Resolution protocol
	[x25|2056]	CCITT .25 protocol
	[xeroxaddrtrans|513]	Xerox PUP Address Translation protocol
	[xeroxpup|512]	Xerox PUP protocol
ethv2	Ethernet Version 2 encapsulation	
nosnapllc	LLC without SNAP encapsulation	
snapllc	LLC SNAP encapsulation	
<1-4094>	Specify a VLAN ID to which an untagged packet is mapped in the range *<1-4094>*	

Mode Global Configuration

Usage If the protocol type matches the protocol specified in the VLAN classifier rule, the received packets are mapped to the specified VLAN. Ethernet Frame Numbers may be entered in place of the protocol names listed. For a full list please refer to the IANA list online: http://www.iana.org/assignments/ethernet-numbers.

Example

```
        awplus# configure terminal

awplus(config)# vlan classifier rule 1 proto x25 encap ethv2
                vlan 2

awplus(config)# vlan classifier rule 2 proto 512 encap ethv2
                vlan 2

awplus(config)# vlan classifier rule 3 proto 2056 encap ethv2
                vlan 2

awplus(config)# vlan classifier rule 4 proto 2054 encap ethv2
                vlan 2
```

Validation Output

```
awplus# show vlan classifier rule
```

```
vlan classifier rule 16 proto rarp encap ethv2 vlan 2
vlan classifier rule 8 proto  encap ethv2 vlan 2
vlan classifier rule 4 proto arp encap ethv2 vlan 2
vlan classifier rule 2 proto xeroxpup encap ethv2 vlan 2
```

Related Commands show vlan classifier rule
vlan classifier activate
vlan classifier group

vlan database

Use this command to enter the VLAN Configuration mode.

Syntax

```
vlan database
```

Mode

Global Configuration

Usage

Use this command to enter the VLAN configuration mode. You can then add or delete a VLAN, or modify its values.

Example

In the following example, note the change to VLAN configuration mode from Configure mode:

```
        awplus# configure terminal
  awplus(config)# vlan database
awplus(config-vlan)#
```

Related Commands

vlan

vlan statistics

This command creates a VLAN packet counter instance, and enables you to add one or more ports to a defined counter instance. This command can only be applied to switch ports. You cannot apply it to aggregated links or eth ports.

The **no** variant of this command enables the deletion of VLAN packet counter instances, or for removing one or more ports that are currently mapped to a counter instance. Note that the selected range of ports must all be switch ports.

Note In describing this command, the terms frame and packet are used interchangeably.

Syntax

```
vlan <vid> statistics name <instance_name>

no vlan statistics name <instance_name>
```

Parameter	Description
<vid>	The VID of the VLAN that is associated with *<instance-name>*.
<instance-name>	The name of the instance for which incoming frames and their bytes are counted.

Mode Interface Configuration

Usage A maximum of 128 packet counter instances can be created. When the first instance is configured, the switch will reserve sufficient resources to support 128 packet counter instances. These resources are also shared with other features such as QoS and ACLs. Where the remaining resources are insufficient to support the VLAN Statistics feature the feature will not be enabled, and an error message will display.

Examples Create a VLAN packet counter instance named **vlan2-data**, and apply this to count incoming vlan2 tagged frames on ports 1.0.4 and 1.0.5.

```
          awplus#  configure terminal
   awplus(config)#  interface port1.0.4,port1.0.5
awplus(config-if)#  vlan 2 statistics name vlan2-data
```

From the previous example, add ports in the range 1.0.2 to 1.0.3 to the VLAN packet counter instance. The **vlan2-data** instance will now count all incoming vlan2 tagged frames on ports within the range 1.0.1 to 1.0.5.

```
   awplus(config)#  interface port1.0.2-port1.0.3
awplus(config-if)#  vlan 2 statistics name vlan2-data
```

To remove `port1.0.5` from the packet counter instance named **vlan2-data.**

```
    awplus(config)# interface port1.0.5
 awplus(config-if)# no vlan statistics name vlan2-data
```

To remove the remaining ports 1.0.2 to 1.0.4 from the packet counter instance named **vlan2-data.** Note that because there are no ports associated with the **vlan2-data**, this instance will be removed.

```
    awplus(config)# interface port1.0.2-port1.0.4
 awplus(config-if)# no vlan statistics name vlan2-data
```

Related Commands clear vlan statistics
show vlan statistics

Chapter 18: Spanning Tree Introduction: STP, RSTP, and MSTP

Introduction

This chapter describes and provides configuration procedures for:

- Spanning Tree Protocol (STP)
- Rapid Spanning Tree Protocol (RSTP)
- Multiple Spanning Tree Protocol (MSTP)

For detailed information about the commands used to configure spanning trees, see Chapter 19, Spanning Tree Commands.

Overview of Spanning Trees

The concept of the spanning tree protocol was devised to address broadcast storming. The spanning tree algorithm itself is defined by the IEEE standard 802.1D and its later revisions.

The IEEE Standard 802.1 uses the term "bridge" to define the spanning tree operation and uses terms such as Bridge Protocol Data Units, Root Bridge etc., when defining spanning tree protocol functions.

When a bridge receives a frame, it reads the source and destination address fields. The bridge then enters the frame's source address in its forwarding database. In doing this the bridge associates the frame's source address with the network attached to the port on which the frame was received. The bridge also reads the destination address and if it can find this address in its forwarding database, it forwards the frame to the appropriate port. If the bridge does not recognize the destination address, it forwards the frame out from all its ports except for the one on which the frame was received, and then waits for a reply. This process is known as "flooding".

A significant problem arises where bridges connect via multiple paths. A frame that arrives with an unknown destination address is flooded over all available paths. The arrival of these frames at another network via different paths and bridges produces major problems. The bridges can become confused about the location of the send and receive devices and begin sending frames in the wrong directions. This process feeds on itself and produces a condition known as a broadcast storm, where the increase of circulating frames can eventually overload the network.

Spanning tree operation

Where a LAN's topology results in more than one path existing between bridges, frames transmitted onto the extended LAN circulate in increasing numbers around the loop, decreasing performance and potentially overloading the network. However, multiple paths through the extended LAN are often required in order to provide redundancy and backup in the event of a bridge or link failure.

The spanning tree is created through the exchange of Bridge Protocol Data Units (BPDUs) between the bridges in the LAN. The spanning tree algorithm operates by:

- Automatically computing a loop-free portion of the topology, called a *spanning tree*. The topology is dynamically pruned to the spanning tree by declaring certain ports on a switch to be redundant, and placing them into a 'Blocking' state.
- Automatically recovering from a switch failure that would partition the extended LAN by reconfiguring the spanning tree to use redundant paths, if available.

The logical tree computed by the spanning tree algorithm has the following properties:

- A single bridge is selected to become the spanning tree's unique *root bridge*. This is the device that advertises the lowest Bridge ID. Each bridge is uniquely identified by its Bridge ID, which comprises the bridge's *root priority* (a spanning tree parameter) followed by its MAC address.
- Each bridge or LAN in the tree, except the root bridge, has a unique parent, known as the *designated bridge.* Each LAN has a single bridge, called the *designated bridge*, that connects it to the next LAN on the path towards the root bridge.
- Each port connecting a bridge to a LAN has an associated *cost*, called the *root path cost.* This is the sum of the costs for each path between the particular bridge port and the root bridge. The designated bridge for a LAN is the one that advertises the lowest *root path cost*. If two bridges on the same LAN have the same lowest root path cost, then the switch with the lowest bridge ID becomes the designated bridge.

The spanning tree computation is a continuous, distributed process to establish and maintain a spanning tree (Table 18-1). The basic algorithm is similar for STP, RSTP and MSTP modes.

Table 18-1: Spanning tree process

The spanning tree algorithm ...	By ...
Selects a root bridge	It selects as the root bridge for the spanning tree the device with the (numerically) lowest bridge identifier (that is, the device with lowest root bridge priority value, or if they have the same priority, the bridge with the lowest MAC address).
Selects root ports	On each device, it selects the root port according to: ■ the port with the lowest path cost to the root bridge ■ the port connected to the bridge with the lowest root identifier ■ MSTP and RSTP only: the port with the lowest port priority value ■ the port with the lowest port number[1]
Blocks alternate ports	In order to prevent loops, it blocks alternate ports (Discarding state) that provide higher cost paths to the root bridge.
Blocks backup ports	Where a second port connects one switch back to itself, it blocks the backup port that has the highest path cost or port number.
Selects designated ports	All other ports that are not disabled are selected as designated ports and are eventually made active (Forwarding state).
Maintains the spanning tree	If a switch or port fails, the spanning tree configures a new active topology, changing some port states, to reestablish connectivity and block loops. Depending on where the failure occurs, the changes may be widespread (e.g., if the root bridge fails), or local (e.g., if a designated port fails).

[1] The whole three part port number (D.M.P) is used to find the lowest port number; where: D is the device number within a stack (1 for a non stacked device), M is the module number (always 0 on the x510 switch), and P is the number of the port within the XEM or base-board.

The logical spanning tree, sometimes called the *active topology*, includes the root bridge and all designated bridges, meaning all ports that are to be used for communication within the spanning tree. These ports are in the forwarding state. Ports removed from the logical spanning tree are not in the forwarding state. To implement the spanning tree algorithm, devices communicate with one another using the Spanning Tree Protocol.

Spanning tree modes

STP can run in one of three modes: STP, RSTP or MSTP. A device running RSTP is compatible with other devices running STP; a device running MSTP is compatible with other devices running RSTP or STP. By default, on a device in MSTP mode each port automatically detects the mode of the device connected to it (MSTP, RSTP or STP), and responds in the appropriate mode by sending messages (BPDUs) in the corresponding format. Ports on a device in RSTP mode can automatically detect and respond to connected devices in RSTP and STP mode. Particular ports can also be forced to only operate in a particular mode (**spanning-tree force-version** command on page 19.42).

STP The Spanning Tree Protocol (STP) is the original protocol defined by IEEE standard 802.1D-1988. It creates a single spanning tree over a network.

STP mode may be useful for supporting applications and protocols whose frames may arrive out of sequence or duplicated, for example NetBeui.

RSTP Rapid Spanning Tree Protocol (RSTP) also creates a single spanning tree over a network. Compared with STP, RSTP provides for more rapid convergence to an active spanning tree topology. RSTP is defined in IEEE standard 802.1D-2004.

By default, the device operates in RSTP mode.

MSTP The Multiple Spanning Tree Protocol (MSTP) addresses the limitations in the previous spanning tree protocols, STP and RSTP, within networks that use multiple VLANs with topologies that employ alternative physical links. It supports multiple spanning tree instances on any given link within a network, and supports large networks by grouping bridges into regions that appear as a single bridge to other devices.

MSTP is defined in IEEE standard 802.1Q-2005. The protocol builds on, and remains compatible with, the previous IEEE standards defining STP and RSTP.

Spanning Tree Protocol (STP)

STP uses the process described in Table 18-1 to avoid loops.

STP port states

In STP mode, each switch port can be in one of five spanning tree states, and one of two switch states. The state of a switch port is taken into account by STP. The STP port states (Table 18-2) affect the behavior of ports whose switch state is enabled.

Table 18-2: STP port states

State	Meaning
DISABLED	STP operations are disabled on the port. The port does not participate in the operation of the Spanning Tree Algorithm and Protocol. The port can still switch if its switch state is enabled.
BLOCKING	The forwarding process discards received frames and does not submit forwarded frames for transmission. This is the "standby" mode.The port does not participate in frame relay.
LISTENING	The port is enabled for receiving frames only. The port is preparing to participate in frame relay. The forwarding process discards received frames and does not submit forwarded frames for transmission.
LEARNING	The port is enabled for receiving frames only, and the Learning Process can add new source address information to the Forwarding Database.
FORWARDING	The normal state for a switch port. The forwarding process and the Spanning Tree entity are enabled for transmit and receive operations on the port.

Configuring STP

By default, RSTP is enabled on all switch ports. This section provides a procedure for configuring STP (Table 18-3).

To configure other modes, see "Configuring RSTP" on page 18.9 or "Configuring MSTP" on page 18.19.

Table 18-3: Configuration procedure for STP

Command	Description
Configure STP RSTP is enabled by default with default settings on all switch ports to prevent Layer 2 loops in your network.	

`awplus#`
`configure terminal`
	Enter Global Configuration mode.

`awplus(config)#`
`spanning-tree mode stp`
	By default, the device is in RSTP mode. Change to STP mode.

`awplus(config)#`
`spanning-tree enable`
	By default, spanning tree is enabled on all switch ports. If it has been disabled, enable it for STP.

`awplus(config)#`
`spanning-tree priority <priority>`
	By default, all devices have the same root bridge priority, 32768 (8000 in hexadecimal), so the device with the lowest MAC address becomes the root bridge. If you want the device to be the root bridge, set the root bridge priority to a value lower than 32768.
	Enter a value in the range 0 to 61440. If you enter a number that is not a multiple of 4096, the switch rounds the number down.

Configure Root Guard The Root Guard feature makes sure that the port on which it is enabled is a designated port. If the Root Guard enabled port receives a superior BPDU, it goes to a Listening state (for STP) or discarding state (for RSTP and MSTP).	

`awplus(config)#`
`interface <port-list>`
	Enter Interface Configuration mode for the switch ports you want to enable Root Guard for.

`awplus(config-if)#`
`spanning-tree guard root`
	Enable the Guard Root feature for these ports.

`awplus(config-if)#`
`exit`
	Return to Global Configuration mode.

`awplus(config)#`
`exit`
	Return to Privileged Exec mode.

Table 18-3: Configuration procedure for STP(cont.)

Check STP configuration	
awplus# `show spanning-tree [interface <port-list>]`	Display the spanning tree configuration for the device, and confirm the new root bridge priority (Bridge Priority). Note that the Bridge ID is in a form like this: 80000000cd240331, and that other IDs follow the same pattern. This is made up of: 8000—the devices' root bridge priority in hexadecimal 0000cd240331—the devices' MAC address.

Advanced configuration: For most networks the default settings for path costs will be suitable, however, you can configure them if required (spanning-tree path-cost).

Rapid Spanning Tree Protocol (RSTP)

RSTP uses the process described in Table 18-1 to avoid loops.

A spanning tree running in STP mode can take up to one minute to rebuild after a topology or configuration change. The RSTP algorithm provides for a faster recovery of connectivity following the failure of a bridge, bridge port, or a LAN. RSTP provides rapid recovery by including port roles in the computation of port states, and by allowing neighboring bridges to explicitly acknowledge signals on a point-to-point link that indicate that a port wants to enter the forwarding mode.

In rapid mode, the rapid transition of a port to the forwarding state is possible when the port is considered to be part of a point-to-point link, or when the port is considered to be an *edge* port. An edge port is one that attaches to a LAN that has no other bridges attached.

Table 18-4: RSTP port states

State	Meaning
DISABLED	STP operations are disabled on the port.
DISCARDING	The port does not participate in frame relay. The forwarding process discards received frames and does not submit forwarded frames for transmission.
LEARNING	The port is enabled for receiving frames only, and the learning process can add new source address information to the forwarding database. The port does not forward any frames.
FORWARDING	The normal state for a switch port. The forwarding process and the Spanning Tree entity are enabled for transmit and receive operations on the port.

Configuring RSTP

RSTP is enabled by default with default settings on all switch ports to prevent Layer 2 loops in your network. No further configuration is required if you want to use RSTP with these default settings. For further RSTP configuration, see Table 18-5 below.

To configure other modes, see "Configuring MSTP" on page 18.19 or "Configuring STP" on page 18.6.

For detailed configuration examples, see the How To Note *How To Configure Basic Switching Functionality*, available from http://www.alliedtelesis.com.

Table 18-5: Configuration procedure for RSTP

Command	Description
Configure RSTP RSTP is enabled by default with default settings on all switch ports to prevent Layer 2 loops in your network. No further configuration is required if you want to use RSTP with these default settings. If you need to restore the device to RSTP after it has been set to another mode, or modify the default RSTP settings, follow the procedure below.	
awplus# configure terminal	Enter Global Configuration mode.
awplus(config)# spanning-tree mode rstp	By default, the device is in RSTP mode. If it has been changed to STP or MSTP mode, change it back to RSTP.
awplus(config)# spanning-tree enable	By default, spanning tree is enabled on all switch ports. If it has been disabled, enable it for RSTP.
awplus(config)# spanning-tree priority <priority>	By default, all devices have the same root bridge priority, 32768 (8000 in hexadecimal), so the device with the lowest MAC address becomes the root bridge. If you want the device to be the root bridge, set the root bridge priority to a value lower than 32768. Enter a value in the range 0 to 61440. If you enter a number that is not a multiple of 4096, the switch rounds the number down.
Configure edge ports If some switch ports are connected to devices that cannot generate BPDUs (such as workstations), you can set particular switch ports as edge ports, or set them to automatically detect whether they are edge ports.	
awplus(config)# interface <port-list>	Enter Interface Configuration mode for these switch ports.
awplus(config-if)# spanning-tree edgeport (RSTP and MSTP) or awplus(config-if)# spanning-tree autoedge (RSTP and MSTP)	Set these ports to be edge ports, or set these ports to automatically detect whether they are edge ports.

Table 18-5: Configuration procedure for RSTP(cont.)

Configure Root Guard	
awplus(config-if)# exit	Return to Global Configuration mode.
awplus(config)# interface *<port-list>*	Enter Interface Configuration mode for the switch ports you want to enable Root Guard for.
awplus(config-if)# spanning-tree guard root	The Root Guard feature makes sure that the port on which it is enabled is a designated port. If the Root Guard enabled port receives a superior BPDU, it goes to a Listening state (for STP) or discarding state (for RSTP and MSTP). Enable the Guard Root feature if required.
Configure BPDU Guard	
awplus(config-if)# exit	Return to Global Configuration mode.
awplus(config)# spanning-tree portfast bpdu-guard	If required, enable the BPDU Guard feature.
awplus(config)# spanning-tree errdisable-timeout enable	Set a timeout for ports that are disabled due to the BPDU guard feature.
awplus(config)# spanning-tree errdisable-timeout interval	Specify the time interval after which a port is brought back up when it has been disabled by the BPDU guard feature.
Check RSTP configuration	
awplus(config)# exit	Return to Privileged Exec mode.
awplus# show spanning-tree [interface <port-list>]	Display the spanning tree configuration for the device, and confirm the new root bridge priority (Bridge Priority). Note that the Bridge ID is in a form like this: 80000000cd240331, and that other IDs follow the same pattern. This is made up of: 8000—the devices' root bridge priority in hexadecimal 0000cd240331—the devices' MAC address.

Advanced configuration: For most networks the default settings for path costs will be suitable, however, you can configure them if required (spanning-tree path-cost).

Multiple Spanning Tree Protocol (MSTP)

Conceptually, MSTP views the total bridged network as one that comprises a number of *Multiple Spanning Tree Regions* (MSTRs), where each region can contain up to 64 spanning trees, which operate locally, called *Multiple Spanning Tree Instances* (MSTIs). AlliedWare Plus^{TM} supports up to 15 MSTIs. The regions are linked by the *Common Internal Spanning Tree* (CIST).

MSTP uses BPDUs to exchange information between spanning-tree compatible devices, to prevent loops in each MSTI and also in the CIST, by selecting active and blocked paths. This process is described in Table 18-1.

If multiple ports are aggregated together into a dynamic (LACP) or static channel group, then the spanning-tree process is aware of the link aggregation and treats the aggregated ports as a single logical path.

Advantage of MSTP over RSTP

MSTP is similar to RSTP, in that it provides loop resolution and rapid convergence. However, RSTP can keep track of only one spanning-tree. MSTP can track many spanning-trees, referred to as *instances*. MSTP makes it possible to have different forwarding paths for different MST instances. This enables load balancing of network traffic across redundant links, so that all the links in a network can be used by at least one MSTI, and no link is left completely idle. That is to say that no link is unnecessarily shut down by spanning-tree.

Essentially, MSTP is VLAN aware and RSTP is not VLAN aware. MSTP BPDUs and RSTP BPDUs are compatible, so a network can have a mixture of MSTP and RSTP areas.

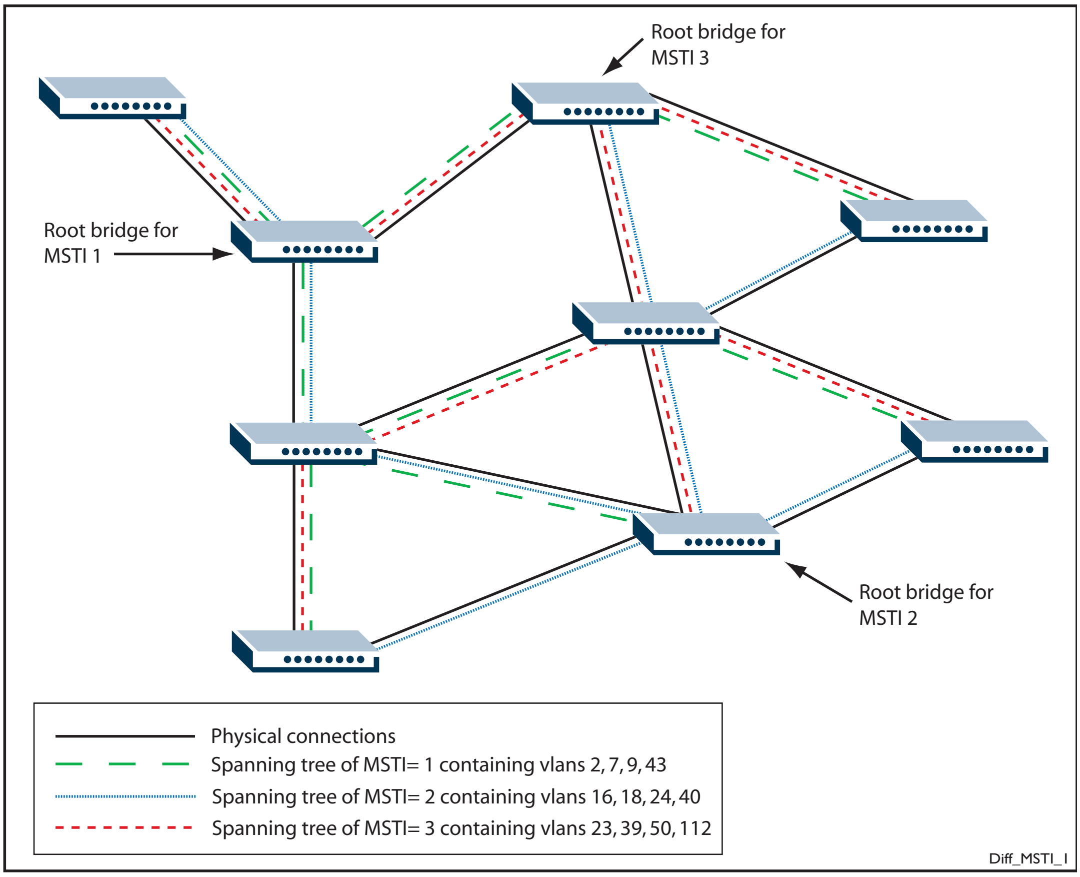
Multiple Spanning Tree Instances (MSTI)

MSTP enables the grouping and mapping of VLANs to different spanning tree instances. So, an MST Instance (MSTI) is a particular set of VLANs that are all using the same spanning tree.

In a network where all VLANs span all links of the network, judicious choice of bridge priorities for different MSTIs can result in different switches becoming root bridges for different MSTIs. That will result in the different MSTIs choosing different active topologies on the network. An example of how different MSTIs can choose different active topologies on the same physical set of links is illustrated in Figure 18-1.

MSTP is compatible with RSTP and STP—see "Common and Internal Spanning Tree (CIST)" on page 18.15.

Figure 18-1: Different spanning trees created by different MSTIs on the same physical layout


Root bridge for
MSTI 3
Root bridge for
MSTI 1
Root bridge for
MSTI 2
Physical connections
Spanning tree of MSTI= 1 containing vlans 2, 7, 9, 43
Spanning tree of MSTI= 2 containing vlans 16, 18, 24, 40
Spanning tree of MSTI= 3 containing vlans 23, 39, 50, 112
Diff_MSTI_1

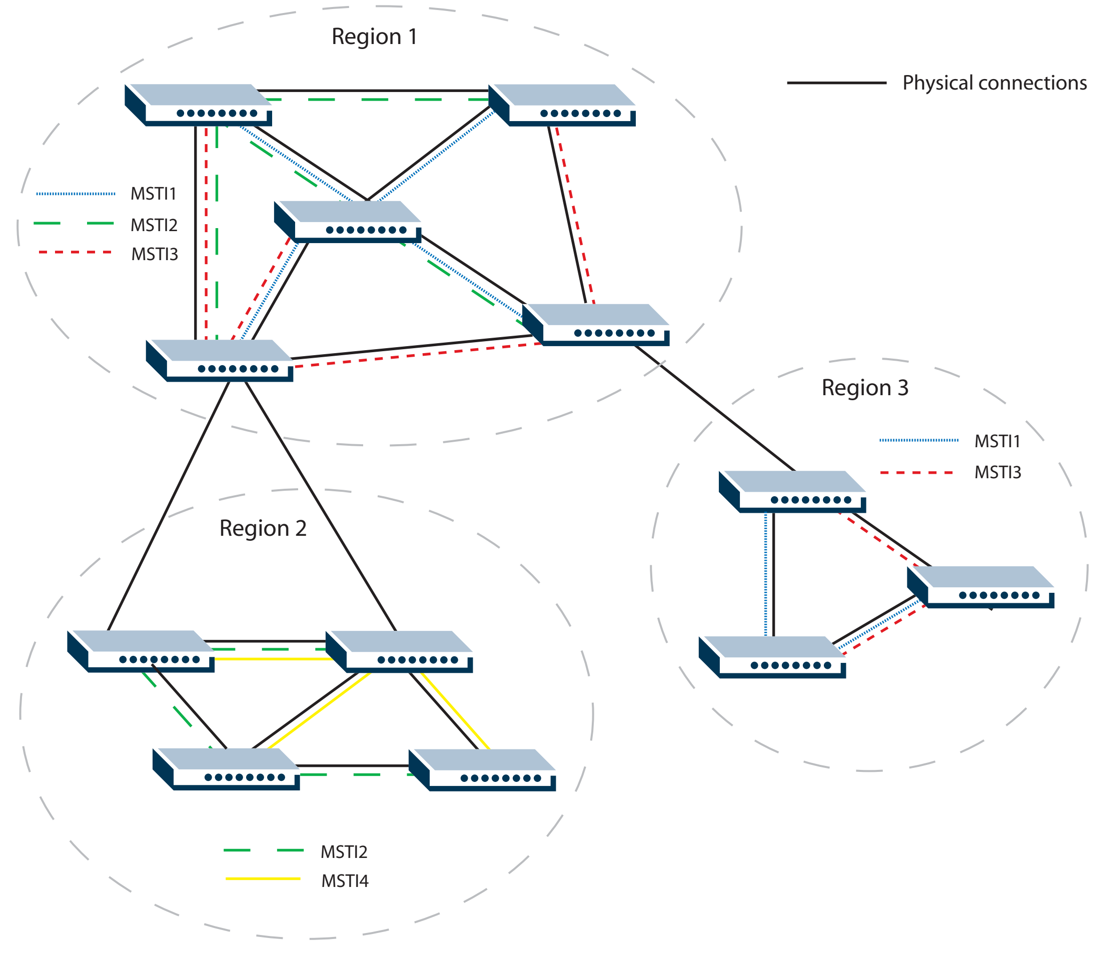
MSTP Regions

An MST region is a set of interconnected switches that all have the same values for the following MST configuration identification elements:

- MST configuration name - the name of the MST region
- Revision level - the revision number of configuration
- Configuration Digest - the mapping of which VLANs are mapped to which MST instances

Each of the MST instances created are identified by an MSTI number. This number is locally significant within the MST region. Therefore, an MSTI will not span across MST regions.

Figure 18-2: MSTIs in different regions



The MSTI1 in Region 1 is unrelated to the MSTI1 in Region 3. Similarly, the MSTI2 in Region 1 is quite unrelated to the MSTI2 in Region 2.

MSTI_numbers

The task of assigning each bridge to a particular region is achieved by the member bridges each comparing their *MST Configuration Identifiers*. More information on configuration identifiers is provided in Table 18-6, but for the moment an *MST Configuration Identifier* can simply be thought of as an identifier that represents the mapping of VLANs to MSTIs within each bridge. Therefore, bridges with identical *MST Configuration Identifiers*, must have identical MSTI mapping tables.

While each MSTI can have multiple VLANs, each VLAN can be associated with only one MSTI. Once these associations have been made, the bridges in each region can transmit their spanning tree BPDUs and advertise their MSTIs. This in turn establishes the active data paths between the bridges for each group of VLANs (that is, for each MSTI) and block any duplicate paths within each instance. A particular advantage of this enhancement applies where a large number of VLANs share a few internetwork paths. In this situation there need only be as many Multiple Spanning Tree Instances (MSTIs) as there are source and destination bridge pairs, remembering that a pair of bridges probably has multiple paths between them.

In order to ensure that each bridge within a region maintains the same configuration information (particularly their VID to MSTI mappings) and to ensure each bridge's membership of a particular region, the bridges exchange configuration information in the form of *MST Configuration Identifiers*. Table 18-6 provides a breakdown of an *MST Configuration Identifier.* A detailed explanation of bridge configuration identifiers can be found in Section 13.7 of the IEEE 802.1Q-2003 standard.

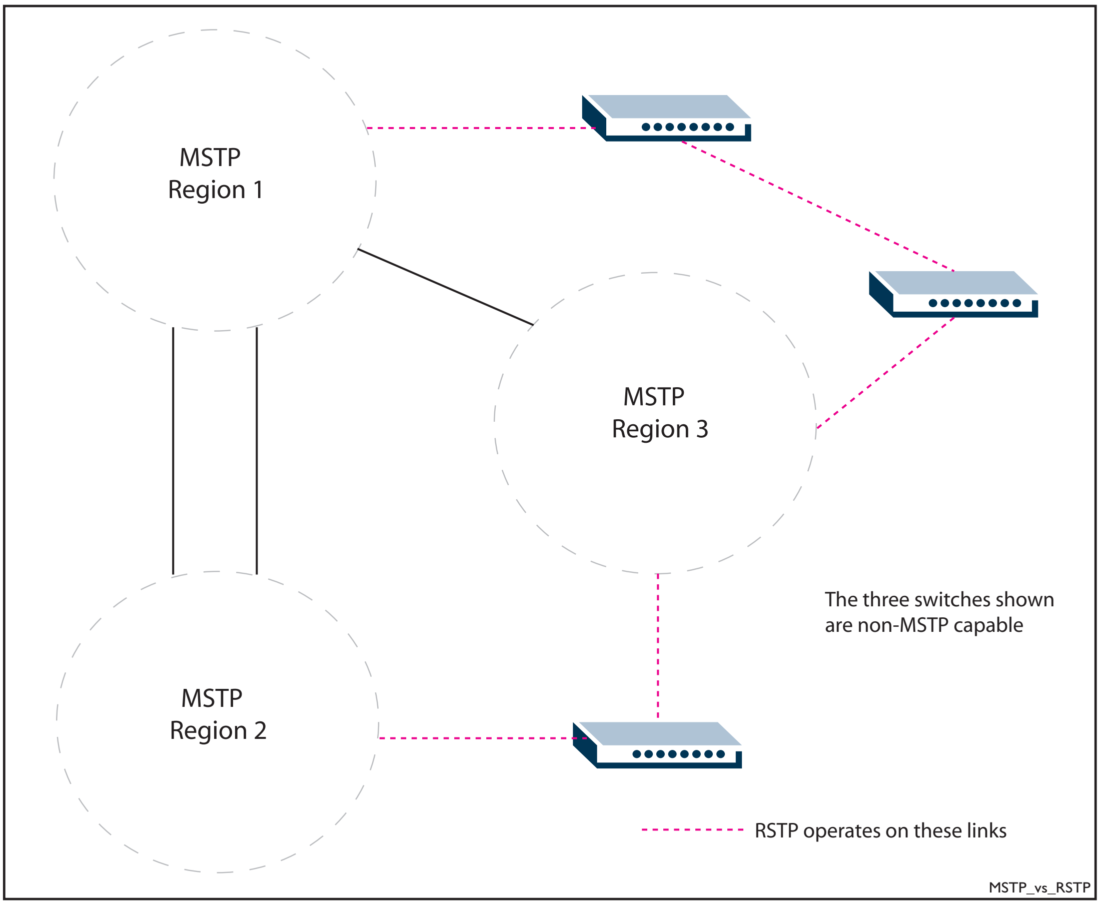
Table 18-6: MST Configuration Identifier

Field Name	**Description**
Format Selector	A single octet field whose value of 0 indicates MSTP operation
Region Name	A name (up to 32 characters long) that identifies a particular MST region, defined using the region (MSTP) command on page 19.12
Revision Level	A number representing the region's revision level, defined using the revision (MSTP) command on page 19.13.
Configuration Digest	A 16 octet (HMAC-MD5 based) signature created from the MST configuration table.

Common and Internal Spanning Tree (CIST)

The CIST is the default spanning tree instance of MSTP, i.e. all VLANs that are not members of particular MSTIs are members of the CIST. Also, an individual MST region can be regarded as a single virtual bridge by other MST regions. The spanning tree that runs between regions is the CIST. The CIST is also the spanning tree that runs between MST regions and Single Spanning Tree (SST) entities. So, in Figure 18-3, the STP that is running between the regions, and to the SST bridges, is the CIST.

Figure 18-3: The CIST operates on links between regions and to SST devices


MSTP
Region 1
MSTP
Region 3
MSTP
Region 2
The three switches shown are non-MSTP capable
RSTP operates on these links
MSTP_vs_RSTP

Compatibility with Previous Spanning Tree Protocols

MSTP provides for compatibility with older spanning tree protocols in several ways. In addition to the MST region described in the previous section, the protocol provides for single spanning tree systems by employing a Common and Internal Spanning Tree (CIST). The CIST applies a common and internal spanning tree protocol to the whole of the bridged network and is a direct equivalent to the internal spanning tree (IST) protocol of earlier versions.

In common with legacy spanning tree systems, the CIST protocol first determines its root bridge from all the bridges on the network. This is the bridge that contains the lowest bridge identifier. The protocol then selects a regional root bridge for each MSTR. This is the bridge that provides the best path to the CIST root. After the MSTR root bridges have been chosen, they then act on the region's behalf in such a way that the region appears to the Common Spanning Tree (CST) as a virtual bridge. So in addition to having multiple MSTIs, each region operates as a bridge in a CST.

CIST

In addition to the individual MSTIs within each MSTP region, the MSTP region is a member of a network-wide spanning tree called the Common and Internal Spanning Tree (CIST). Conceptually, each region represents a virtual bridge. Internal and external bridge connectivity are two independent functions.

Frames with VIDs allocated to the CIST are subject to the rules and path costs of the complete bridged LAN as determined by the CIST's vectors. Frames other than these are subject to the CIST when travelling outside their region, and subject to its particular MSTI inside the region.

The following operational rules apply:

- Each bridge can be a member of only one region.
- A data frame is associated with a single VID.
- Data frames with a given VID are associated with either the CIST or their particular MSTI, but not both.

The role of the Common Spanning Tree (CST) in a network, and the Common and Internal Spanning Tree (CIST) configured on each device, is to prevent loops within a wider network that may span more than one MSTP region and parts of the network running in legacy STP or RSTP mode.

CIST first allocates root and designated bridges by selecting the bridge with the lowest identifier as the root. MSTP then deals with any loops between the regions in the CST. It does this by considering the CIST "vectors" in the following order:

1. CIST External Root Path Cost
2. CIST Regional Root Identifier
3. CIST Internal Root Path Cost
4. CIST Designated Bridge Identifier
5. CIST Designated Port Identifier
6. CIST Receiving Port Identifier

MSTP Bridge Protocol Data Units (BPDUs)

The main function of bridge protocol data units is to enable MSTP to select its root bridges for the CIST ("Common and Internal Spanning Tree (CIST)" on page 18.15) and each MSTI. MSTP is compatible with earlier spanning tree versions; its Bridge Protocol Data Unit (BPDU) formats build on earlier versions ("Compatibility with Previous Spanning Tree Protocols" on page 18.15).

Table 18-7 shows the standardized format for MSTP BPDU messages. The general format of the BPDUs comprise a common generic portion—octets 1 to 36—that are based on those defined in IEEE Standard 802.1D, 1998, followed by components that are specific to CIST—octets 37 to 102. Components specific to each MSTI are added to this BPDU data block.

Table 18-7: MSTP Bridge Protocol Data Units (BPDUs)

Field Name	Octets	Description
Protocol Identifier	1–2	Protocol being used. The value 0000 0000 0000 0000 identifies the spanning tree algorithm and protocol.
Protocol Version Identifier	3	Identifies the protocol version used.
BPDU Type	4	Value 0000 0000 specifies a configuration BPDU.
CIST Flags	5	Bit 1 is the topology change flag. Bit 2 conveys the CIST proposal flag in RST and MST BPDUs - unused in STP. Bits 3 & 4 convey the CIST port role in RST, and MST BPDUs - unused in STP. Bit 5 conveys the CIST learning flag in RST and MST BPDUs - unused in STP. Bit 6 conveys the CIST forwarding flag in RST and MST BPDUs - unused in STP. Bit 7 conveys the CIST agreement flag in RST and MST BPDUs - unused in STP. Bit 8 conveys the topology change acknowledge flag in STP configuration BPDUs - unused in RSTP and MSTP BPDUs.
CIST Root Identifier	6–13	The Bridge identifier of the CIST Root
CIST External Path Cost	14–17	The path cost between MST regions from the transmitting bridge to the CIST root.
CIST Regional Root Identifier	18–25	ID of the current CIST regional root bridge.
CIST Port Identifier	26–27	CIST port identifier of the transmitting bridge port.
Message Age	28–29	Message age timer value.
Max Age	30–31	Timeout value to be used by all bridges in the bridged network. This value is set by the root. Some implementations of MSTP may choose not to use this value.
Hello Time	32–33	Time interval between the generation of configuration BPDUs by the root bridge.
Forward Delay	34–35	A timeout value used to ensure forward delay timer consistency when transferring a port to the forwarding state. It is also used for ageing filtering database dynamic entries following changes in the active topology.

Table 18-7: MSTP Bridge Protocol Data Units (BPDUs)(cont.)

Field Name	Octets	Description
Version 1 Length	36	Used to convey the Version 1 length. It is always transmitted as 0.
Version 3 Length	37–38	Used to convey the Version 3 length. It is the number of octets taken by the parameters that follow in the BPDU.
MST Configuration Identifier	39–89	An identifier comprising elements of the following: Format Selector Configuration Name Revision Level Configuration Digest.
CIST Internal Root Path Cost	90–93	Path cost to the CIST regional root.
CIST Bridge Identifier	94–101	CIST bridge identifier of the transmitting bridge.
CIST Remaining Hops	102	Remaining hops which limits the propagation and longevity of received spanning tree information for the CIST.
MSTI Configuration Messages (may be absent)	103–39 plus Version 3 Length	See Table 18-8.

Table 18-8: MSTI configuration messages

Field Name	Octets	Description
MSTI Flags	1	Bits 1 through 8, convey the topology change flag, proposal flag, port role (two bits), Learning flag, forwarding flag, agreement flag, and master flag for this MSTI.
MSTI Regional Root Identifier	2–9	This includes the value of the MSTID for this configuration message encoded in bits 4 through 1 of octet 1, and bits 8 through 1 of octet 2.
MSTI Internal Root Path Cost	10-13	Internal Root Path Cost.
MSTI Bridge Priority	14	Bits 5 through 8 convey the value of the bridge identifier priority for this MSTI. Bits 1 through 4 of Octet 14 are transmitted as 0, and ignored on receipt.
MSTI Port Priority	15	Bits 5 through 8 are used to convey the value of the port identifier priority for this MSTI. Bits 1 through 4 are transmitted as 0, and ignored on receipt.
MSTI Remaining Hops	16	Value of remaining hops for this MSTI.

Configuring MSTP

By default, RSTP is enabled with default settings on all switch ports. To configure MSTP, see the configuration procedure in Table 18-9.

To configure other modes, see "Configuring RSTP" on page 18.9 or "Configuring STP" on page 18.6.

For detailed configuration examples, see the How To Note *How To Configure Basic Switching Functionality,* available from website at http://www.alliedtelesis.com.

Configuration guidelines for MSTP

- Switches must have the same MST configuration identification elements (region name, revision level and VLAN to MSTI mapping) to be in the same MST region. When configuring multiple MST regions for MSTP, MSTIs are locally significant within an MST region. MSTIs will not span from one region to another region.
- Common and Internal Spanning Tree (CIST) is the default spanning tree instance for MSTP. This means that all VLANs that are not explicitly configured into another MSTI are members of the CIST.
- The software supports a single instance of the MSTP Algorithm consisting of the CIST and up to 15 MSTIs.
- A VLAN can only be mapped to one MSTI or to the CIST. One VLAN mapped to multiple spanning trees is not allowed. All the VLANs are mapped to the CIST by default. Once a VLAN is mapped to a specified MSTI, it is removed from the CIST.
- An MSTI is locally significant within an MST region. An MSTI cannot span across multiple MST regions. The CIST is the spanning tree instance for connecting different MST regions and single spanning tree entities, such as RSTP and STP switches.
- MSTP is compatible with RSTP and STP. An MST region appears as a virtual bridge connecting to single spanning tree entities.
- To avoid unnecessary STP processing, a port that attaches to a LAN that is known to have no other bridges/switches attached can be configured as an edge port.

Before configuring MSTP

Before configuring MSTP, configure VLANs and associate them with switch ports (Chapter 16, VLAN Introduction and Chapter 17, VLAN Commands), and determine for your network:

- which MSTP regions, revision level and instances are required
- which VLANs and switch ports will belong to which MSTIs,
- which devices you want to be root bridges for each MSTI

Table 18-9: Configuration procedure for MSTP

Command	Description

`awplus#`
`configure terminal`
	Enter Global Configuration mode.

`awplus(config)#`
`spanning-tree mode mstp`
	By default, the device is in RSTP mode. Change to MSTP mode.

`awplus(config)#`
`spanning-tree enable`
	By default, spanning tree is enabled on all switch ports. If it has been disabled, enable it for MSTP.

Table 18-9: Configuration procedure for MSTP(cont.)

Configure MSTP region, revision, and instances All MSTP devices in this region of the network must have the same region name, revision number, and VLAN to MSTI mappings.	

`awplus(config)#`
`spanning-tree mst configuration`
	Enter MST Configuration mode.

`awplus(config-mst)#`
`region <region-name>`
	Specify the MSTP region. The **region-name** parameter is an arbitrary string that specifies the name you want to assign to the MST region for identification.

`awplus(config-mst)#`
`revision <revision-number>`
	The **revision-number** parameter specifies the revision of the current MST configuration. The revision is an arbitrary number that you assign to an MST region. It can be used to keep track of the number of times that MST configuration has been updated for the network.
	Specify the MST revision number in the range 0 to 255.

`awplus(config-mst)#`
`instance <msti-id> vlan {<vid>| <vid-list>}`
	To allow MSTP to block traffic for different VLANs in different places in a loop, create multiple MSTP instances and associate VLANs with them. Each VLAN can only be in one instance.
	Specify the MST instance ID in the range 1 to 15.

Advanced configuration The commands above are the minimum required to configure MSTP. The following commands allow more advanced configuration.	
Assign root bridge priorities MSTP lets you distribute traffic more efficiently across a network by blocking different links for different VLANs. You do this by making different devices into the root bridge for each MSTP instance, and for the CIST, so that each instance blocks a different link. By default, all devices have the same root bridge priority, 32768 (8000 in hexadecimal), so the device with the lowest MAC address becomes the root bridge. If you want the device to be the root bridge for an instance or for the CIST, set the priority to a lower value (a higher priority) than other devices for this instance. (If you enter a number that is not a multiple of 4096, the device rounds the number down.)	

`awplus(config)#`
`spanning-tree mst configuration`
	Enter MST Configuration mode.

`awplus(config-mst)#`
`instance <msti-id> priority <priority>`
	Set the priority for the device to become the root bridge for each instance.
	Specify the MST instance ID in the range 1 to 15.
	Specify the root bridge priority in the range 0 to 61440. If you enter a number that is not a multiple of 4096, the switch rounds the number down.

`awplus(config-mst)#`
`exit`
	Return to Global Configuration mode.

Table 18-9: Configuration procedure for MSTP(cont.)

awplus(config)# `spanning-tree priority <priority>`	Set the priority for the device to become the root bridge for the CIST. Specify the bridge priority in the range 0 to 61440. If you enter a number that is not a multiple of 4096, the switch rounds the number down.
Configure edge ports If some switch ports are connected to devices that cannot generate BPDUs (such as workstations), you can set particular switch ports as edge ports, or set them to automatically detect whether they are edge ports.	
awplus(config)# `interface <port-list>`	Enter Interface Configuration mode for these switch ports.

awplus(config-if)#
`spanning-tree edgeport (RSTP and MSTP)`
or
awplus(config-if)#
`spanning-tree autoedge (RSTP and MSTP)`
	Set these ports to be edge ports,
	or
	set these ports to automatically detect whether they are edge ports.

Configure Root Guard	
awplus(config-if)# `spanning-tree guard root`	The Root Guard feature makes sure that the port on which it is enabled is a designated port. If the Root Guard enabled port receives a superior BPDU, it goes to a Listening state (for STP) or discarding state (for RSTP and MSTP). Enable the Guard Root feature if required.
awplus(config-if)# `exit`	Return to Global Configuration mode.
Configure BPDU Guard	
awplus(config)# `spanning-tree portfast bpdu-guard`	If required, enable the BPDU Guard feature.
awplus(config)# `spanning-tree errdisable-timeout enable`	Set a timeout for ports that are disabled due to the BPDU guard feature.
awplus(config)# `spanning-tree errdisable-timeout interval <10-1000000>`	Specify the time interval after which a port is brought back up when it has been disabled by the BPDU guard feature.

Table 18-9: Configuration procedure for MSTP(cont.)

Check MSTP configuration	
awplus(config)# exit	Return to Privileged Exec mode.
awplus# show spanning-tree mst config	Check that the digest is the same on this device as for all other devices in the same region.
awplus# show spanning-tree mst	Check the MST to VLAN and port mapping.
awplus# show spanning-tree mst instance <instance>	Check the detailed information for a particular instance, and all switch ports associated with that instance. Specify the MST instance ID in the range 1 to 15.
awplus# show spanning-tree mst interface <port>	Check general information about MSTP, and the CIST settings.

Advanced configuration: For most networks, the default settings of the following will be suitable. However, you can also configure them.

- path costs for ports in an MSTI (spanning-tree mst instance path-cost) or for the CIST (spanning-tree path-cost)
- port priority for ports in an MSTI (spanning-tree mst instance priority) or for the CIST (spanning-tree priority (port priority))

Chapter 19: Spanning Tree Commands

Command List

This chapter provides an alphabetical reference for commands used to configure RSTP, STP or MSTP. For information about spanning trees, including configuration procedures, see Chapter 18, Spanning Tree Introduction: STP, RSTP, and MSTP

clear spanning-tree statistics

Use this command to clear all the STP BPDU (Bridge Protocol Data Unit) statistics.

Syntax

```
clear spanning-tree statistics

clear spanning-tree statistics [instance <mstp-instance>]

clear spanning-tree statistics
   [interface <port> [instance <mstp-instance>]]
```

Parameter	Description
<port>	The port to clear STP BPDU statistics for. The port may be a switch port (e.g. `port1.0.4`), a static channel group (e.g. `sa3`), or a dynamic (LACP) channel group (e.g. `po4`).
<mstp-instance>	The MSTP instance (MSTI - Multiple Spanning Tree Instance) to clear MSTP BPDU statistics.

Mode User Exec and Privileged Exec

Usage Use this command with the **instance** parameter in MSTP mode. Specifying this command with the **interface** parameter only not the instance parameter will work in STP and RSTP mode.

Examples

```
awplus# clear spanning-tree statistics
```

```
awplus# clear spanning-tree statistics instance 1
```

```
awplus# clear spanning-tree statistics interface port1.0.2
```

```
awplus# clear spanning-tree statistics interface port1.0.2
        instance 1
```

clear spanning-tree detected protocols (RSTP and MSTP)

Use this command to clear the detected protocols for a specific port, or all ports.

Use this command in RSTP or MSTP mode only.

Syntax

```
clear spanning-tree detected protocols [interface <port>]
```

Parameter	Description
<port>	The port to clear detected protocols for. The port may be a switch port (e.g. `port1.0.4`), a static channel group (e.g. `sa3`), or a dynamic (LACP) channel group (e.g. `po4`).

Mode Privileged Exec

Example

```
awplus# clear spanning-tree detected protocols
```

debug mstp (RSTP and STP)

Use this command to enable debugging for the configured spanning tree mode, and echo data to the console, at various levels. Note that although this command uses the keyword **mstp** it displays debugging output for RSTP and STP protocols as well the MSTP protocol.

Use the **no** variant of this command to disable spanning tree debugging.

Syntax

```
debug mstp {all|cli|protocol [detail]|timer [detail]}

debug mstp {packet {rx|tx} [decode] [interface <interface>]}

debug mstp {topology-change [interface <interface>]}

no debug mstp {all|cli|protocol [detail]|timer [detail]}

no debug mstp {packet {rx|tx} [decode] [interface <interface>]}

no debug mstp {topology-change [interface <interface>]}
```

Parameter	Description
`all`	Echoes all spanning tree debugging levels to the console.
`cli`	Echoes spanning tree commands to the console.
`packet`	Echoes spanning tree packets to the console.
`rx`	Received packets.
`tx`	Transmitted packets.
`protocol`	Echoes protocol changes to the console.
`timer`	Echoes timer information to the console.
`detail`	Detailed output.
`decode`	Interprets packet contents
`topology-change`	Interprets topology change messages
`interface`	Keyword before <*interface*> placeholder to specify an interface to debug
`<interface>`	Placeholder used to specify the name of the interface to debug.

Mode Privileged Exec and Global Configuration mode

Usage 1 Use the **debug mstp topology-change interface** command to generate debugging messages when the switch receives an indication of a topology change in a BPDU from another device. The debugging can be activated on a per-port basis. Although this command uses the keyword **mstp**, it displays debugging output for RSTP and STP protocols as well as the MSTP protocol.

Due to the likely volume of output, these debug messages are best viewed using the terminal monitor command on page 8.65 before issuing the relevant **debug mstp** command. The default terminal monitor filter will select and display these messages. Alternatively, the messages can be directed to any of the other log outputs by adding a filter for the MSTP application using log buffered (filter) command on page 10.9:

```
awplus# configure terminal
awplus(config)# log buffered program mstp
```

Output 1

```
awplus#terminal monitor
awplus#debug mstp topology-change interface port1.0.19
10:09:09 awplus MSTP[1409]: Topology change rcvd on port1.0.19 (internal)
10:09:09 awplus MSTP[1409]: Topology change rcvd on MSTI 1 port1.0.19
aawplus#debug mstp topology-change interface port1.0.21
10:09:29 awplus MSTP[1409]: Topology change rcvd on port1.0.21 (external)
10:09:29 awplus MSTP[1409]: Topology change rcvd on MSTI 1 port1.0.21
```

Usage 2 Use the **debug mstp packet rx|tx decode interface** command to generate debugging messages containing the entire contents of a BPDU displayed in readable text for transmitted and received xSTP BPDUs. The debugging can be activated on a per-port basis and transmit and receive debugging is controlled independently. Although this command uses the keyword **mstp**, it displays debugging output for RSTP and STP protocols as well as the MSTP protocol.

Due to the likely volume of output, these debug messages are best viewed using the terminal monitor command on page 8.65 before issuing the relevant **debug mstp** command. The default terminal monitor filter will select and display these messages. Alternatively, the messages can be directed to any of the other log outputs by adding a filter for the MSTP application using the log buffered (filter) command on page 10.9:

```
awplus(config)# log buffered program mstp
```

Output 2 In MSTP mode - an MSTP BPDU with 1 MSTI:

```
awplus#terminal monitor
awplus#debug mstp packet rx decode interface port1.0.19
17:23:42 awplus MSTP[1417]: port1.0.19 xSTP BPDU rx - start
17:23:42 awplus MSTP[1417]: Protocol version: MSTP, BPDU type: RST
17:23:42 awplus MSTP[1417]: CIST Flags:  Agree Forward Learn role=Desig
17:23:42 awplus MSTP[1417]: CIST root id      : 0000:0000cd1000fe
17:23:42 awplus MSTP[1417]: CIST ext pathcost : 0
17:23:42 awplus MSTP[1417]: CIST reg root id  : 0000:0000cd1000fe
17:23:42 awplus MSTP[1417]: CIST port id      : 8001 (128:1)
17:23:42 awplus MSTP[1417]: msg age: 0 max age: 20 hellotime: 2 fwd delay: 15
17:23:42 awplus MSTP[1417]: Version 3 length : 80
17:23:42 awplus MSTP[1417]: Format id      : 0
17:23:42 awplus MSTP[1417]: Config name    : test
17:23:42 awplus MSTP[1417]: Revision level : 0
17:23:42 awplus MSTP[1417]: Config digest  : 3ab68794d602fdf43b21c0b37ac3bca8
17:23:42 awplus MSTP[1417]: CIST int pathcost : 0
17:23:42 awplus MSTP[1417]: CIST bridge id    : 0000:0000cd1000fe
17:23:42 awplus MSTP[1417]: CIST hops remaining : 20
17:23:42 awplus MSTP[1417]: MSTI flags           : Agree Forward Learn role=Desig
17:23:42 awplus MSTP[1417]: MSTI reg root id     : 8001:0000cd1000fe
17:23:42 awplus MSTP[1417]: MSTI pathcost        : 0
17:23:42 awplus MSTP[1417]: MSTI bridge priority : 32768 port priority : 128
17:23:42 awplus MSTP[1417]: MSTI hops remaining  : 20
17:23:42 awplus MSTP[1417]: port1.0.19 xSTP BPDU rx - finish
```

In STP mode transmitting a TCN BPDU:

```
awplus#terminal monitor
awplus#debug mstp packet tx decode interface port1.0.19
17:28:09 awplus MSTP[1417]: port1.0.19 xSTP BPDU tx - start
17:28:09 awplus MSTP[1417]: Protocol version: STP, BPDU type: TCN
17:28:09 awplus MSTP[1417]: port1.0.19 xSTP BPDU tx - finish
```

In STP mode receiving an STP BPDU:

```
awplus#terminal monitor
awplus#debug mstp packet rx decode interface port1.0.19
17:31:36 awplus MSTP[1417]: port1.0.19 xSTP BPDU rx - start
17:31:36 awplus MSTP[1417]: Protocol version: STP, BPDU type: Config
17:31:36 awplus MSTP[1417]: Flags:  role=none
17:31:36 awplus MSTP[1417]: Root id       : 8000:0000cd1000fe
17:31:36 awplus MSTP[1417]: Root pathcost : 0
17:31:36 awplus MSTP[1417]: Bridge id   : 8000:0000cd1000fe
17:31:36 awplus MSTP[1417]: Port id       : 8001 (128:1)
17:31:36 awplus MSTP[1417]: msg age: 0 max age: 20 hellotime: 2 fwd delay: 15
17:31:36 awplus MSTP[1417]: ort1.0.19 xSTP BPDU rx - finish
```

In RSTP mode receiving an RSTP BPDU:

```
awplus#terminal monitor
awplus#debug mstp packet rx decode interface port1.0.19
awplus#17:30:17 awplus MSTP[1417]: port1.0.19 xSTP BPDU rx - start
17:30:17 awplus MSTP[1417]: Protocol version: RSTP, BPDU type: RST
17:30:17 awplus MSTP[1417]: CIST Flags:  Forward Learn role=Desig
17:30:17 awplus MSTP[1417]: CIST root id      : 8000:0000cd1000fe
17:30:17 awplus MSTP[1417]: CIST ext pathcost : 0
17:30:17 awplus MSTP[1417]: CIST reg root id  : 8000:0000cd1000fe
17:30:17 awplus MSTP[1417]: CIST port id      : 8001 (128:1)
17:30:17 awplus MSTP[1417]: msg age: 0 max age: 20 hellotime: 2 fwd delay: 15
17:30:17 awplus MSTP[1417]: port1.0.19 xSTP BPDU rx - finish
```

Examples

```
awplus# debug mstp all
```

```
awplus# debug mstp cli
```

```
awplus# debug mstp packet rx
```

```
awplus# debug mstp protocol detail
```

```
awplus# debug mstp timer
```

```
awplus# debug mstp packet rx decode interface port1.0.2
```

```
awplus# debug mstp packet tx decode interface port1.0.12
```

Related commands log buffered (filter)
show debugging mstp
terminal monitor
undebug mstp

instance priority (MSTP)

Use this command to set the priority for this device to become the root bridge for the specified MSTI (Multiple Spanning Tree Instance).

Use this command for MSTP only.

Use the **no** variant of this command to restore the root bridge priority of the device for the instance to the default.

Syntax

```
instance <msti-id> priority <priority>

no instance <msti-id> priority
```

Parameter	Description
<msti-id>	Specify the The MST instance ID in the range <1-63>.
<priority>	Specify the root bridge priority for the device for the MSTI in the range <0-61440>. Note that a lower priority number indicates a greater likelihood of the device becoming the root bridge. The priority values can be set only in increments of 4096. If you specify a number that is not a multiple of 4096, it will be rounded down. The default priority is 32768.

Default The default priority value for all instances is 32768.

Mode MST Configuration Mode

Usage MSTP lets you distribute traffic more efficiently across a network by blocking different links for different VLANs. You do this by making different devices into the root bridge for each MSTP instance, so that each instance blocks a different link.

If all devices have the same root bridge priority for the instance, MSTP selects the device with the lowest MAC address to be the root bridge. Give the device a higher priority for becoming the root bridge for a particular instance by assigning it a lower priority number, or vice versa.

Examples To set the root bridge priority for MSTP instance 2 to be the highest (0), so that it will be the root bridge for this instance when available, use the commands:

```
          awplus# configure terminal
  awplus(config)# spanning-tree mst configuration
awplus(config-mst)# instance 2 priority 0
```

To reset the root bridge priority for instance 2 to the default (32768), use the commands:

```
          awplus# configure terminal
  awplus(config)# spanning-tree mst configuration
awplus(config-mst)# no instance 2 priority
```

Related Commands region (MSTP)
revision (MSTP)
show spanning-tree mst config
spanning-tree mst instance
spanning-tree mst instance priority

instance vlan (MSTP)

Use this command to create an MST Instance (MSTI), and associate the specified VLANs with it. An MSTI is a spanning tree instance that exists within an MST region (MSTR). An MSTR can contain up to 63 MSTIs.

When a VLAN is associated with an MSTI the member ports of the VLAN are automatically configured to send and receive spanning-tree information for the associated MSTI. You can disable this automatic configuration of member ports of the VLAN to the associated MSTI by using a **no spanning-tree mst instance** command to remove the member port from the MSTI.

Use the **instance vlan** command for MSTP only.

Use the **no** variant of this command to remove the specified VLANs from the MSTI.

Syntax

```
instance <msti-id> vlan {<vid>|<vid-list>}

no instance <msti-id> vlan {<vid>|<vid-list>}
```

Parameter	Description
<msti-id>	Specify the MST instance ID <1-63>.
<vid>	Specify a VLAN identifier (VID) in the range <1-4094> to be associated with the MSTI specified.
<vid-list>	A hyphen-separated range or a comma-separated list of VLAN IDs

Mode MST Configuration mode

Usage The VLANs must be created before being associated with an MST instance (MSTI). If the VLAN range is not specified, the MSTI will not be created.

This command removes the specified VLANs from the CIST and adds them to the specified MSTI. If you use the **no** variant of this command to remove the VLAN from the MSTI, it returns it to the CIST. To move a VLAN from one MSTI to another, you must first use the **no** variant of this command to return it to the CIST.

Ports in these VLANs will remain in the control of the CIST until you associate the ports with the MSTI using the spanning-tree mst instance command.

Example

```
          awplus# configure terminal
  awplus(config)# spanning-tree mode mstp
  awplus(config)# spanning-tree mst configuration
awplus(config-mst)# instance 2 vlan 30
```

Related Commands

region (MSTP)
revision (MSTP)
show spanning-tree mst config
spanning-tree mst instance
vlan

region (MSTP)

Use this command to assign a name to the device's MST Region. MST Instances (MSTI) of a region form different spanning trees for different VLANs.

Use this command for MSTP only.

Use the **no** variant of this command to remove this region name and reset it to the default.

Syntax

```
region <region-name>

no region
```

Parameter	Description
<region-name>	Specify the name of the region, up to 32 characters. Valid characters are upper-case, lower-case, digits, underscore.

Default By default, the region name is My Name.

Mode MST Configuration mode

Usage The region name, the revision number, and the digest of the VLAN to MSTI configuration table must be the same on all devices that are intended to be in the same MST region.

Example

```
           awplus# configure terminal
   awplus(config)# spanning-tree mst configuration
awplus(config-mst)# region ATL
```

Related Commands revision (MSTP)
show spanning-tree mst config

revision (MSTP)

Use this command to specify the MST revision number to be used in the configuration identifier.

Use this command for MSTP only.

Syntax

```
revision <revision-number>
```

Parameter	Description
<revision-number>	<0-255> Revision number.

Default The default of revision number is 0.

Mode MST Configuration Mode

Usage The region name, the revision number, and the digest of the VLAN to MSTI configuration table must be the same on all devices that are intended to be in the same MST region.

Example

```
           awplus# configure terminal
   awplus(config)# spanning-tree mst configuration
awplus(config-mst)# revision 25
```

Related Commands

region (MSTP)
show spanning-tree mst config
instance vlan (MSTP)

show debugging mstp

Use this command to show the MSTP debugging options set.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax

```
show debugging mstp
```

Mode

User Exec and Privileged Exec mode

Example

To display the MSTP debugging options set, enter the command:

```
awplus# show debugging mstp
```

Output

Figure 19-1: Example output from the **show debugging mstp** command

```
MSTP debugging status:
  MSTP receiving packet debugging is on
```

Related Commands

debug mstp (RSTP and STP)

show spanning-tree

Use this command to display detailed spanning tree information on the specified port or on all ports. Use this command for RSTP, MSTP or STP.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax

```
show spanning-tree [interface <port-list>]
```

Parameter	Description
`interface`	Display information about the following port only.
<port-list>	The ports to display information about. A port-list can be: ■ a switch port (e.g. `port1.0.12`) a static channel group (e.g. `sa3`) or a dynamic (LACP) channel group (e.g. `po3`) ■ a continuous range of ports separated by a hyphen, e.g. `port1.0.1-1.0.24`, or `sa1-2`, or `po1-4` ■ a comma-separated list of ports and port ranges, e.g. `port1.0.1,port1.0.4-1.2.24`. Do not mix switch ports, static channel groups, and dynamic (LACP) channel groups in the same list

Mode User Exec, Privileged Exec and Interface Configuration mode

Usage Note that any list of interfaces specified must not span any interfaces that are not installed.

A topology change counter has been included for RSTP and MSTP. You can see the topology change counter for RSTP by using the **show spanning-tree** command. You can see the topology change counter for MSTP by using the **show spanning-tree mst instance** command.

Example To display spanning tree information about `port1.0.23`, use the command:

```
awplus# show spanning-tree interface port1.0.23
```

Output Figure 19-2: Example output from the **show spanning-tree** command

```
% 1: Bridge up - Spanning Tree Enabled
% 1: Root Path Cost 0 - Root Port 0 -  Bridge Priority 32768
% 1: Forward Delay 15 - Hello Time 2 - Max Age 20
% 1: Root Id 80000000cd20f093
% 1: Bridge Id 80000000cd20f093
% 1: last topology change Sun Nov 20 12:24:24 1977
% 1: portfast bpdu-filter disabled
% 1: portfast bpdu-guard disabled
% 1: portfast errdisable timeout disabled
% 1: portfast errdisable timeout interval 300 sec
%   port1.0.23: Port 5023 - Id 839f - Role Designated - State Forwarding
%   port1.0.23: Designated Path Cost 0
%   port1.0.23: Configured Path Cost 200000  - Add type Explicit ref count 1
%   port1.0.23: Designated Port Id 839f - Priority 128  -
%   port1.0.23: Root 80000000cd20f093
%   port1.0.23: Designated Bridge 80000000cd20f093
%   port1.0.23: Message Age 0 - Max Age 20
%   port1.0.23: Hello Time 2 - Forward Delay 15
%   port1.0.23: Forward Timer 0 - Msg Age Timer 0 - Hello Timer 1 - topo change
timer 0
%   port1.0.23: forward-transitions 32
%   port1.0.23: Version Rapid Spanning Tree Protocol - Received None - Send RSTP
%   port1.0.23: No portfast configured - Current  portfast off
%   port1.0.23: portfast bpdu-guard  default  - Current portfast bpdu-guard off
%   port1.0.23: portfast bpdu-filter default  - Current portfast bpdu-filter off
%   port1.0.23: no root guard configured     - Current root guard off
%   port1.0.23: Configured Link Type point-to-point - Current point-to-point
.
.
```

The following example output is for the **show spanning-tree** command in RSTP mode.

Figure 19-3: Example output from the **show spanning-tree** command

```
awplus#show spanning-tree
% 1: Bridge up - Spanning Tree Enabled
% 1: Root Path Cost 0 - Root Port 0 -  Bridge Priority 32768
% 1: Forward Delay 15 - Hello Time 2 - Max Age 20
% 1: Root Id 80000000cd24ff2d
% 1: Bridge Id 80000000cd24ff2d
% 1: last topology change Thu Jul 26 02:06:26 2007
% 1: portfast bpdu-filter disabled
% 1: portfast bpdu-guard disabled
% 1: portfast errdisable timeout disabled
% 1: portfast errdisable timeout interval 300 sec
%   port1.0.1: Port 5001 - Id 8389 - Role Disabled - State Discarding
%   port1.0.1: Designated Path Cost 0
%   port1.0.1: Configured Path Cost 20000000  - Add type Explicit ref count 1
%   port1.0.1: Designated Port Id 8389 - Priority 128  -
%   port1.0.1: Root 80000000cd24ff2d
%   port1.0.1: Designated Bridge 80000000cd24ff2d
%   port1.0.1: Message Age 0 - Max Age 20
%   port1.0.1: Hello Time 2 - Forward Delay 15
%   port1.0.1: Forward Timer 0 - Msg Age Timer 0 - Hello Timer 0 - topo change
timer 0
%   port1.0.1: forward-transitions 0
%   port1.0.1: Version Rapid Spanning Tree Protocol - Received None - Send STP
%   port1.0.1: No portfast configured - Current  portfast off
%   port1.0.1: portfast bpdu-guard  default   - Current portfast bpdu-guard off
%   port1.0.1: portfast bpdu-filter default   - Current portfast bpdu-filter off
%   port1.0.1: no root guard configured     - Current root guard off
%   port1.0.1: Configured Link Type point-to-point - Current  shared
%
%   port1.0.2: Port 5002 - Id 838a - Role Disabled - State Discarding
%   port1.0.2: Designated Path Cost 0
%   port1.0.2: Configured Path Cost 20000000  - Add type Explicit ref count 1
%   port1.0.2: Designated Port Id 838a - Priority 128  -
%   port1.0.2: Root 80000000cd24ff2d
%   port1.0.2: Designated Bridge 80000000cd24ff2d
%   port1.0.2: Message Age 0 - Max Age 20
%   port1.0.2: Hello Time 2 - Forward Delay 15
%   port1.0.2: Forward Timer 0 - Msg Age Timer 0 - Hello Timer 0 - topo change
timer 0
%   port1.0.2: forward-transitions 0
%   port1.0.2: Version Rapid Spanning Tree Protocol - Received None - Send STP
%   port1.0.2: No portfast configured - Current  portfast off
%   port1.0.2: portfast bpdu-guard  default   - Current portfast bpdu-guard off
%   port1.0.2: portfast bpdu-filter default   - Current portfast bpdu-filter off
%   port1.0.2: no root guard configured     - Current root guard off
%   port1.0.2: Configured Link Type point-to-point - Current  shared
%
```

show spanning-tree brief

Use this command to display a summary of spanning tree status information on all ports. Use this command for RSTP, MSTP or STP.

Syntax

```
show spanning-tree brief
```

Parameter	Description
`brief`	A brief summary of spanning tree information.

Mode User Exec, Privileged Exec and Interface Configuration

Usage Note that any list of interfaces specified must not span any interfaces that are not installed.

A topology change counter has been included for RSTP and MSTP. You can see the topology change counter for RSTP by using the **show spanning-tree** command. You can see the topology change counter for MSTP by using the **show spanning-tree mst instance** command.

Example To display a summary of spanning tree status information, use the command:

```
awplus# show spanning-tree brief
```

Output Figure 19-4: Example output from the **show spanning-tree brief** command

```
Default: Bridge up - Spanning Tree Enabled
Default: Root Path Cost 40000 - Root Port 4501 -  Bridge Priority 32768
Default: Root Id 8000:0000cd250001
Default: Bridge Id 8000:0000cd296eb1

Port            Designated Bridge    Port Id    Role          State
sa1             8000:001577c9744b    8195       Rootport      Forwarding
po1             8000:0000cd296eb1    81f9       Designated    Forwarding
port1.0.1       8000:0000cd296eb1    8389       Disabled      Discarding
port1.0.2       8000:0000cd296eb1    838a       Disabled      Discarding
port1.0.3       8000:0000cd296eb1    838b       Disabled      Discarding
port1.0.4       8000:0000cd296eb1    838c       Disabled      Discarding
port1.0.5       8000:0000cd296eb1    838d       Disabled      Discarding
port1.0.6       8000:0000cd296eb1    838e       Disabled      Discarding
port1.0.9       8000:0000cd296eb1    8391       Disabled      Discarding
port1.0.10      8000:0000cd296eb1    8392       Disabled      Discarding
port1.0.11      8000:0000cd296eb1    8393       Disabled      Discarding
port1.0.12      8000:0000cd296eb1    8394       Disabled      Discarding%
```

Related Commands show spanning-tree

show spanning-tree mst

This command displays bridge-level information about the CIST and VLAN to MSTI mappings.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax

```
show spanning-tree mst
```

Mode User Exec, Privileged Exec and Interface Configuration

Example To display bridge-level information about the CIST and VLAN to MSTI mappings, enter the command:

```
awplus# show spanning-tree mst
```

Output Figure 19-5: Example output from the **show spanning-tree mst** command

```
% 1: Bridge up - Spanning Tree Enabled
% 1: CIST Root Path Cost 0 - CIST Root Port 0 - CIST Bridge
Priority 32768
% 1: Forward Delay 15 - Hello Time 2 - Max Age 20 - Max-hops 20
% 1: CIST Root Id 8000000475e93ffe
% 1: CIST Reg Root Id 8000000475e93ffe
% 1: CST Bridge Id 8000000475e93ffe
% 1: portfast bpdu-filter disabled
% 1: portfast bpdu-guard disabled
% 1: portfast errdisable timeout disabled
% 1: portfast errdisable timeout interval 300 sec
%
%   Instance        VLAN
%   0:              1
%   2:              4
```

Related Commands show spanning-tree mst interface

show spanning-tree mst config

Use this command to display MSTP configuration identifier for the device.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax `show spanning-tree mst config`

Mode User Exec, Privileged Exec and Interface Configuration

Usage The region name, the revision number, and the digest of the VLAN to MSTI configuration table must be the same on all devices that are intended to be in the same MST region.

Example To display MSTP configuration identifier information, enter the command:

```
awplus# show spanning-tree mst config
```

Output Figure 19-6: Example output from the **show spanning-tree mst config** command

```
awplus#show spanning-tree mst config
%
%  MSTP Configuration Information:
%-------------------------------------------------------
%  Format Id      : 0
%  Name           : My Name
%  Revision Level : 0
%  Digest         : 0x80DEE46DA92A98CF21C603291B22880A
%-------------------------------------------------------
```

Related Commands

instance vlan (MSTP)
region (MSTP)
revision (MSTP)

show spanning-tree mst detail

This command displays detailed information about each instance, and all interfaces associated with that particular instance.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax

```
show spanning-tree mst detail
```

Mode User Exec, Privileged Exec and Interface Configuration

Example To display detailed information about each instance, and all interfaces associated with them, enter the command:

```
awplus# show spanning-tree mst detail
```

Output Figure 19-7: Example output from the **show spanning-tree mst detail** command

```
% 1: Bridge up - Spanning Tree Enabled
% 1: CIST Root Path Cost 0 - CIST Root Port 0 -  CIST Bridge Priority 32768
% 1: Forward Delay 15 - Hello Time 2 - Max Age 20 - Max-hops 20
% 1: CIST Root Id 80000000cd24ff2d
% 1: CIST Reg Root Id 80000000cd24ff2d
% 1: CIST Bridge Id 80000000cd24ff2d
% 1: portfast bpdu-filter disabled
% 1: portfast bpdu-guard disabled
% 1: portfast errdisable timeout disabled
% 1: portfast errdisable timeout interval 300 sec
%   port1.0.1: Port 5001 - Id 8389 - Role Disabled - State Discarding
%   port1.0.1: Designated External Path Cost 0 -Internal Path Cost 0
%   port1.0.1: Configured Path Cost 20000000  - Add type Explicit ref count 1
%   port1.0.1: Designated Port Id 8389 - CIST Priority 128  -
%   port1.0.1: CIST Root 80000000cd24ff2d
%   port1.0.1: Regional Root 80000000cd24ff2d
%   port1.0.1: Designated Bridge 80000000cd24ff2d
%   port1.0.1: Message Age 0 - Max Age 20
%   port1.0.1: CIST Hello Time 2 - Forward Delay 15
%   port1.0.1: CIST Forward Timer 0 - Msg Age Timer 0 - Hello Timer 0 - topo 
change timer 0
.
.
%   port1.0.2: forward-transitions 0
%   port1.0.2: Version Multiple Spanning Tree Protocol - Received None - Send STP
%   port1.0.2: No portfast configured - Current  portfast off
%   port1.0.2: portfast bpdu-guard  default  - Current portfast bpdu-guard off
%   port1.0.2: portfast bpdu-filter default  - Current portfast bpdu-filter off
%   port1.0.2: no root guard configured     - Current root guard off
%   port1.0.2: Configured Link Type point-to-point - Current  shared
%
%   port1.0.3: Port 5003 - Id 838b - Role Disabled - State Discarding
%   port1.0.3: Designated External Path Cost 0 -Internal Path Cost 0
%   port1.0.3: Configured Path Cost 20000000  - Add type Explicit ref count 1
%   port1.0.3: Designated Port Id 838b - CIST Priority 128  -
%   port1.0.3: CIST Root 80000000cd24ff2d
%   port1.0.3: Regional Root 80000000cd24ff2d
%   port1.0.3: Designated Bridge 80000000cd24ff2d
%   port1.0.3: Message Age 0 - Max Age 20
%   port1.0.3: CIST Hello Time 2 - Forward Delay 15
%   port1.0.3: CIST Forward Timer 0 - Msg Age Timer 0 - Hello Timer 0 - topo 
change timer 0
%   port1.0.3: forward-transitions 0
%   port1.0.3: Version Multiple Spanning Tree Protocol - Received None - Send STP
%   port1.0.3: No portfast configured - Current  portfast off
%   port1.0.3: portfast bpdu-guard  default  - Current portfast bpdu-guard off
%   port1.0.3: portfast bpdu-filter default  - Current portfast bpdu-filter off
%   port1.0.3: no root guard configured     - Current root guard off
%   port1.0.3: Configured Link Type point-to-point - Current  shared
```

show spanning-tree mst detail interface

This command prints detailed information about the specified switch port, and the MST instances associated with it.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax

```
show spanning-tree mst detail interface <port>
```

Parameter	Description
<port>	The port to display information about. The port may be a switch port (e.g. `port1.0.4`), a static channel group (e.g. `sa3`), or a dynamic (LACP) channel group (e.g. `po4`).

Mode User Exec, Privileged Exec and Interface Configuration

Example To display detailed information about `port1.0.3` and the instances associated with it, enter the command:

```
awplus# show spanning-tree mst detail interface port1.0.3
```

Output Figure 19-8: Example output from the **show spanning-tree mst detail interface** command

```
% 1: Bridge up - Spanning Tree Enabled
% 1: CIST Root Path Cost 0 - CIST Root Port 0 -  CIST Bridge Priority 32768
% 1: Forward Delay 15 - Hello Time 2 - Max Age 20 - Max-hops 20
% 1: CIST Root Id 80000000cd24ff2d
% 1: CIST Reg Root Id 80000000cd24ff2d
% 1: CIST Bridge Id 80000000cd24ff2d
% 1: portfast bpdu-filter disabled
% 1: portfast bpdu-guard disabled
% 1: portfast errdisable timeout disabled
% 1: portfast errdisable timeout interval 300 sec
%   port1.0.2: Port 5002 - Id 838a - Role Disabled - State Discarding
%   port1.0.2: Designated External Path Cost 0 -Internal Path Cost 0
%   port1.0.2: Configured Path Cost 20000000  - Add type Explicit ref count 2
%   port1.0.2: Designated Port Id 838a - CIST Priority 128  -
%   port1.0.2: CIST Root 80000000cd24ff2d
%   port1.0.2: Regional Root 80000000cd24ff2d
%   port1.0.2: Designated Bridge 80000000cd24ff2d
%   port1.0.2: Message Age 0 - Max Age 20
%   port1.0.2: CIST Hello Time 2 - Forward Delay 15
%   port1.0.2: CIST Forward Timer 0 - Msg Age Timer 0 - Hello Timer 0 - topo
change timer 0
%   port1.0.2: forward-transitions 0
%   port1.0.2: Version Multiple Spanning Tree Protocol - Received None - Send STP
%   port1.0.2: No portfast configured - Current  portfast off
%   port1.0.2: portfast bpdu-guard  default  - Current portfast bpdu-guard off
%   port1.0.2: portfast bpdu-filter default  - Current portfast bpdu-filter off
%   port1.0.2: no root guard configured     - Current root guard off
%   port1.0.2: Configured Link Type point-to-point - Current  shared
%
% Instance  2:  Vlans: 2
% 1: MSTI Root Path Cost 0 -MSTI Root Port 0 - MSTI Bridge Priority 32768
% 1: MSTI Root Id 80020000cd24ff2d
% 1: MSTI Bridge Id 80020000cd24ff2d
%   port1.0.2: Port 5002 - Id 838a - Role Disabled - State Discarding
%   port1.0.2: Designated Internal Path Cost 0  - Designated Port Id 838a
%   port1.0.2: Configured Internal Path Cost 20000000
%   port1.0.2: Configured CST External Path cost 20000000
%   port1.0.2: CST Priority 128  - MSTI Priority 128
%   port1.0.2: Designated Root 80020000cd24ff2d
%   port1.0.2: Designated Bridge 80020000cd24ff2d
%   port1.0.2: Message Age 0 - Max Age 0
%   port1.0.2: Hello Time 2 - Forward Delay 15
%   port1.0.2: Forward Timer 0 - Msg Age Timer 0 - Hello Timer 0
```

show spanning-tree mst instance

This command displays detailed information for the specified instance, and all switch ports associated with that instance.

A topology change counter has been included for RSTP and MSTP. You can see the topology change counter for RSTP by using the show spanning-tree command. You can see the topology change counter for MSTP by using the **show spanning-tree mst instance** command.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax

```
show spanning-tree mst instance <instance>
```

Parameter	Description
<instance>	Specify an MSTP instance in the range <1-63>.

Mode User Exec, Privileged Exec, and Interface Configuration

Usage To display detailed information for **instance 2**, and all switch ports associated with that instance, use the command:

```
awplus# show spanning-tree mst instance 2
```

Output Figure 19-9: Example output from the **show spanning-tree mst instance** command

```
% 1: MSTI Root Path Cost 0 - MSTI Root Port 0 - MSTI Bridge Priority 32768
% 1: MSTI Root Id 80020000cd24ff2d
% 1: MSTI Bridge Id 80020000cd24ff2d
%   port1.0.2: Port 5002 - Id 838a - Role Disabled - State Discarding
%   port1.0.2: Designated Internal Path Cost 0  - Designated Port Id 838a
%   port1.0.2: Configured Internal Path Cost 20000000
%   port1.0.2: Configured CST External Path cost 20000000
%   port1.0.2: CST Priority 128  - MSTI Priority 128
%   port1.0.2: Designated Root 80020000cd24ff2d
%   port1.0.2: Designated Bridge 80020000cd24ff2d
%   port1.0.2: Message Age 0 - Max Age 0
%   port1.0.2: Hello Time 2 - Forward Delay 15
%   port1.0.2: Forward Timer 0 - Msg Age Timer 0 - Hello Timer 0
%
```

show spanning-tree mst instance interface

This command displays detailed information for the specified MST (Multiple Spanning Tree) instance, and the specified switch port associated with that MST instance.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax `show spanning-tree mst instance <instance> interface <port>`

Parameter	Description
<instance>	Specify an MSTP instance in the range <1-63>.
<port>	The port to display information about. The port may be a switch port (e.g. `port1.0.4`), a static channel group (e.g. `sa3`), or a dynamic (LACP) channel group (e.g. `po4`).

Mode User Exec, Privileged Exec, and Interface Configuration

Example To display detailed information for `instance 2`, interface `port1.0.2`, use the command

```
awplus# show spanning-tree mst instance 2 interface port1.0.2
```

Output Figure 19-10: Example output from the **show spanning-tree mst instance** command

```
% 1: MSTI Root Path Cost 0 - MSTI Root Port 0 - MSTI Bridge Priority 32768
% 1: MSTI Root Id 80020000cd24ff2d
% 1: MSTI Bridge Id 80020000cd24ff2d
%   port1.0.2: Port 5002 - Id 838a - Role Disabled - State Discarding
%   port1.0.2: Designated Internal Path Cost 0  - Designated Port Id 838a
%   port1.0.2: Configured Internal Path Cost 20000000
%   port1.0.2: Configured CST External Path cost 20000000
%   port1.0.2: CST Priority 128  - MSTI Priority 128
%   port1.0.2: Designated Root 80020000cd24ff2d
%   port1.0.2: Designated Bridge 80020000cd24ff2d
%   port1.0.2: Message Age 0 - Max Age 0
%   port1.0.2: Hello Time 2 - Forward Delay 15
%   port1.0.2: Forward Timer 0 - Msg Age Timer 0 - Hello Timer 0
%
```

show spanning-tree mst interface

This command displays the number of instances created, and VLANs associated with it for the specified switch port.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax `show spanning-tree mst interface <port>`

Parameter	Description
<port>	The port to display information about. The port may be a switch port (e.g. `port1.0.4`), a static channel group (e.g. `sa3`), or a dynamic (LACP) channel group (e.g. `po4`).

Mode User Exec, Privileged Exec, and Interface Configuration

Example To display detailed information about each instance, and all interfaces associated with them, for `port1.0.4`, use the command:

```
awplus# show spanning-tree mst interface port1.0.4
```

Output Figure 19-11: Example output from the **show spanning-tree mst interface** command

```
% 1: Bridge up - Spanning Tree Enabled
% 1: CIST Root Path Cost 0 - CIST Root Port 0 - CIST Bridge Priority 32768
% 1: Forward Delay 15 - Hello Time 2 - Max Age 20 - Max-hops 20
% 1: CIST Root Id 80000008c73a2b22
% 1: CIST Reg Root Id 80000008c73a2b22
% 1: CST Bridge Id 80000008c73a2b22
% 1: portfast bpdu-filter disabled
% 1: portfast bpdu-guard disabled
% 1: portfast errdisable timeout disabled
% 1: portfast errdisable timeout interval 1 sec
%
%   Instance        VLAN
%   0:              1
%   1:              2-3
%   2:              4-5
```

show spanning-tree mst detail interface

This command displays detailed information about the specified switch port, and the MST instances associated with it.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax

```
show spanning-tree mst detail interface <port>
```

Parameter	Description
<port>	The port to display information about. The port may be a switch port (e.g. `port1.0.4`), a static channel group (e.g. `sa3`), or a dynamic (LACP) channel group (e.g. `po4`).

Mode User Exec, Privileged Exec and Interface Configuration

Example To display detailed information about `port1.0.3` and the instances associated with it, enter the command:

```
awplus# show spanning-tree mst detail interface port1.0.3
```

Output Figure 19-12: Example output from the **show spanning-tree mst detail interface** command

```
% 1: Bridge up - Spanning Tree Enabled
% 1: CIST Root Path Cost 0 - CIST Root Port 0 -  CIST Bridge Priority 32768
% 1: Forward Delay 15 - Hello Time 2 - Max Age 20 - Max-hops 20
% 1: CIST Root Id 80000000cd24ff2d
% 1: CIST Reg Root Id 80000000cd24ff2d
% 1: CIST Bridge Id 80000000cd24ff2d
% 1: portfast bpdu-filter disabled
% 1: portfast bpdu-guard disabled
% 1: portfast errdisable timeout disabled
% 1: portfast errdisable timeout interval 300 sec
%   port1.0.2: Port 5002 - Id 838a - Role Disabled - State Discarding
%   port1.0.2: Designated External Path Cost 0 -Internal Path Cost 0
%   port1.0.2: Configured Path Cost 20000000  - Add type Explicit ref count 2
%   port1.0.2: Designated Port Id 838a - CIST Priority 128  -
%   port1.0.2: CIST Root 80000000cd24ff2d
%   port1.0.2: Regional Root 80000000cd24ff2d
%   port1.0.2: Designated Bridge 80000000cd24ff2d
%   port1.0.2: Message Age 0 - Max Age 20
%   port1.0.2: CIST Hello Time 2 - Forward Delay 15
%   port1.0.2: CIST Forward Timer 0 - Msg Age Timer 0 - Hello Timer 0 - topo
change timer 0
%   port1.0.2: forward-transitions 0
%   port1.0.2: Version Multiple Spanning Tree Protocol - Received None - Send STP
%   port1.0.2: No portfast configured - Current  portfast off
%   port1.0.2: portfast bpdu-guard  default  - Current portfast bpdu-guard off
%   port1.0.2: portfast bpdu-filter default  - Current portfast bpdu-filter off
%   port1.0.2: no root guard configured     - Current root guard off
%   port1.0.2: Configured Link Type point-to-point - Current  shared
%
% Instance  2:  Vlans: 2
% 1: MSTI Root Path Cost 0 -MSTI Root Port 0 - MSTI Bridge Priority 32768
% 1: MSTI Root Id 80020000cd24ff2d
% 1: MSTI Bridge Id 80020000cd24ff2d
%   port1.0.2: Port 5002 - Id 838a - Role Disabled - State Discarding
%   port1.0.2: Designated Internal Path Cost 0  - Designated Port Id 838a
%   port1.0.2: Configured Internal Path Cost 20000000
%   port1.0.2: Configured CST External Path cost 20000000
%   port1.0.2: CST Priority 128  - MSTI Priority 128
%   port1.0.2: Designated Root 80020000cd24ff2d
%   port1.0.2: Designated Bridge 80020000cd24ff2d
%   port1.0.2: Message Age 0 - Max Age 0
%   port1.0.2: Hello Time 2 - Forward Delay 15
%   port1.0.2: Forward Timer 0 - Msg Age Timer 0 - Hello Timer 0
```

show spanning-tree statistics

This command displays BPDU (Bridge Protocol Data Unit) statistics for all spanning-tree instances, and all switch ports associated with all spanning-tree instances. For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax `show spanning-tree statistics`

Mode Privileged Exec

Usage To display BPDU statistics for all spanning-tree instances, and all switch ports associated with all spanning-tree instances, use the command:

```
awplus# show spanning-tree statistics
```

Output Figure 19-13: Example output from the **show spanning-tree statistics** command

```
Port number = 915 Interface = port1.0.11
                ================================
% BPDU Related Parameters
% -----------------------
% Port Spanning Tree                 : Disable
% Spanning Tree Type                 : Rapid Spanning Tree Protocol
% Current Port State                 : Discarding
% Port ID                            : 8393
% Port Number                        : 393
% Path Cost                          : 20000000
% Message Age                        : 0
% Designated Root                    : ec:cd:6d:20:c0:ed
% Designated Cost                    : 0
% Designated Bridge                  : ec:cd:6d:20:c0:ed
% Designated Port Id                 : 8393
% Top Change Ack                     : FALSE
% Config Pending                     : FALSE
% PORT Based Information & Statistics
% -----------------------------------
% Config Bpdu's xmitted              : 0
% Config Bpdu's received             : 0
% TCN Bpdu's xmitted                 : 0
% TCN Bpdu's received                : 0
% Forward Trans Count                : 0
% STATUS of Port Timers
% ---------------------
% Hello Time Configured              : 2
% Hello timer                        : INACTIVE
% Hello Time Value                   : 0
% Forward Delay Timer                : INACTIVE
% Forward Delay Timer Value          : 0
% Message Age Timer                  : INACTIVE
% Message Age Timer Value            : 0
% Topology Change Timer              : INACTIVE
% Topology Change Timer Value        : 0
% Hold Timer                         : INACTIVE
% Hold Timer Value                   : 0
% Other Port-Specific Info
  ------------------------
% Max Age Transitions                : 1
% Msg Age Expiry                     : 0
% Similar BPDUS Rcvd                 : 0
% Src Mac Count                      : 0
% Total Src Mac Rcvd                 : 0
% Next State                         : Learning
% Topology Change Time               : 0
```

show spanning-tree statistics instance

This command displays BPDU (Bridge Protocol Data Unit) statistics for the specified MST (Multiple Spanning Tree) instance, and all switch ports associated with that MST instance. For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax `show spanning-tree statistics instance <instance>`

Parameter	Description
<instance>	Specify an MSTP instance in the range <1-63>.

Mode Privileged Exec

Usage To display BPDU statistics information for MST instance 2, and all switch ports associated with that MST instance, use the command:

```
awplus# show spanning-tree statistics instance 2
```

Output Figure 19-14: Example output from the **show spanning-tree statistics instance** command:

```
% % INST_PORT port1.0.3 Information & Statistics
% ----------------------------------------
% Config Bpdu's xmitted (port/inst)     : (0/0)
% Config Bpdu's received (port/inst)    : (0/0)
% TCN Bpdu's xmitted (port/inst)        : (0/0)
% TCN Bpdu's received (port/inst)       : (0/0)
% Message Age(port/Inst)                : (0/0)
% port1.0.3: Forward Transitions                   : 0
% Next State                            : Learning
% Topology Change Time                  : 0
% INST_PORT port1.0.4 Information & Statistics
% ----------------------------------------
% Config Bpdu's xmitted (port/inst)     : (0/0)
% Config Bpdu's received (port/inst)    : (0/0)
% TCN Bpdu's xmitted (port/inst)        : (0/0)
% TCN Bpdu's received (port/inst)       : (0/0)
% Message Age(port/Inst)                : (0/0)
% port1.0.4: Forward Transitions                   : 0
% Next State                            : Learning
% Topology Change Time                  : 0
% INST_PORT port1.0.5 Information & Statistics
% ----------------------------------------
% Config Bpdu's xmitted (port/inst)     : (0/0)
% Config Bpdu's received (port/inst)    : (0/0)
% TCN Bpdu's xmitted (port/inst)        : (0/0)
% TCN Bpdu's received (port/inst)       : (0/0)
% Message Age(port/Inst)                : (0/0)
% port1.0.5: Forward Transitions                   : 0
% Next State                            : Learning
% Topology Change Time                  : 0%
```

Related commands show spanning-tree statistics

show spanning-tree statistics instance interface

This command displays BPDU (Bridge Protocol Data Unit) statistics for the specified MST (Multiple Spanning Tree) instance and the specified switch port associated with that MST instance.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax `show spanning-tree statistics instance <instance> interface <port>`

Parameter	Description
<instance>	Specify an MSTP instance in the range <1-63>.
<port>	The port to display information about. The port may be a switch port (e.g. port1.0.4), a static channel group (e.g. sa3), or a dynamic (LACP) channel group (e.g. po4).

Mode Privileged Exec

Example To display BPDU statistics for MST instance 2, interface port1.0.2, use the command

```
awplus# show spanning-tree statistics instance 2 interface
        port1.0.2
```

Output Figure 19-15: Example output from the **show spanning-tree statistics instance interface** command

```
awplus#sh spanning-tree statistics interface port1.0.2 instance 1
        Spanning Tree Enabled for Instance : 1
        ==================================
% INST_PORT port1.0.2 Information & Statistics
% ----------------------------------------
% Config Bpdu's xmitted (port/inst)     : (0/0)
% Config Bpdu's received (port/inst)    : (0/0)
% TCN Bpdu's xmitted (port/inst)        : (0/0)
% TCN Bpdu's received (port/inst)       : (0/0)
% Message Age(port/Inst)                : (0/0)
% port1.0.2: Forward Transitions                  :  0
% Next State                            : Learning
% Topology Change Time                  : 0

% Other Inst/Vlan Information & Statistics
% ----------------------------------------
% Bridge Priority                      : 0
% Bridge Mac Address                   : ec:cd:6d:20:c0:ed
% Topology Change Initiator            : 5023
% Last Topology Change Occured         : Mon Aug 22 05:42:06 2011
% Topology Change                      : FALSE
% Topology Change Detected             : FALSE
% Topology Change Count                : 1
% Topology Change Last Recvd from      : 00:00:00:00:00:00
```

Related commands show spanning-tree statistics

show spanning-tree statistics interface

This command displays BPDU (Bridge Protocol Data Unit) statistics for the specified switch port, and all MST instances associated with that switch port.

For information on output options, see “Controlling “show” Command Output” on page 1.41.

Syntax

```
show spanning-tree statistics interface <port>
```

Parameter	Description
<port>	The port to display information about. The port may be a switch port (e.g. `port1.0.4`), a static channel group (e.g. `sa3`), or a dynamic (LACP) channel group (e.g. `po4`).

Mode Privileged Exec

Example To display BPDU statistics about each MST instance for `port1.0.4`, use the command:

```
awplus# show spanning-tree statistics interface port1.0.4
```

Output Figure 19-16: Example output from the **show spanning-tree statistics interface** command

```
awplus#show spanning-tree statistics interface port1.0.2
                Port number = 906 Interface = port1.0.2
                ================================
% BPDU Related Parameters
% -----------------------
% Port Spanning Tree                 : Disable
% Spanning Tree Type                 : Multiple Spanning Tree Protocol
% Current Port State                 : Discarding
% Port ID                            : 838a
% Port Number                        : 38a
% Path Cost                          : 20000000
% Message Age                        : 0
% Designated Root                    : ec:cd:6d:20:c0:ed
% Designated Cost                    : 0
% Designated Bridge                  : ec:cd:6d:20:c0:ed
% Designated Port Id                 : 838a
% Top Change Ack                     : FALSE
% Config Pending                     : FALSE

% PORT Based Information & Statistics
% -----------------------------------
% Config Bpdu's xmitted              : 0
% Config Bpdu's received             : 0
% TCN Bpdu's xmitted                 : 0
% TCN Bpdu's received                : 0
% Forward Trans Count                : 0

% STATUS of Port Timers
% ---------------------
% Hello Time Configured              : 2
% Hello timer                        : INACTIVE
% Hello Time Value                   : 0
% Forward Delay Timer                : INACTIVE
% Forward Delay Timer Value          : 0
% Message Age Timer                  : INACTIVE
% Message Age Timer Value            : 0
% Topology Change Timer              : INACTIVE
% Topology Change Timer Value        : 0
% Hold Timer                         : INACTIVE
% Hold Timer Value                   : 0

% Other Port-Specific Info
  ------------------------
% Max Age Transitions                : 1
% Msg Age Expiry                     : 0
% Similar BPDUS Rcvd                 : 0
% Src Mac Count                      : 0
% Total Src Mac Rcvd                 : 0
% Next State                         : Learning
% Topology Change Time               : 0


% Other Bridge information & Statistics
  --------------------------------------
% STP Multicast Address              : 01:80:c2:00:00:00
% Bridge Priority                    : 32768
% Bridge Mac Address                 : ec:cd:6d:20:c0:ed
% Bridge Hello Time                  : 2
% Bridge Forward Delay               : 15
% Topology Change Initiator          : 5023
% Last Topology Change Occured       : Mon Aug 22 05:41:20 2011
% Topology Change                    : FALSE
% Topology Change Detected           : TRUE
% Topology Change Count              : 1
% Topology Change Last Recvd from    : 00:00:00:00:00:00
```

Related commands show spanning-tree statistics

show spanning-tree vlan range-index

Use this command to display information about MST (Multiple Spanning Tree) instances and the VLANs associated with them including the VLAN range-index value for the switch.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax

```
show spanning-tree vlan range-index
```

Mode Privileged Exec

Example To display information about MST instances and the VLANs associated with them for the switch, including the VLAN range-index value, use the following command:

```
awplus# show spanning-tree vlan range-index
```

Output Figure 19-17: Example output from the **show spanning-tree vlan range-index** command

```
awplus#show spanning-tree vlan range-index
% MST Instance  VLAN        RangeIdx
%      1         1           1
%
```

Related commands show spanning-tree statistics

spanning-tree autoedge (RSTP and MSTP)

Use this command to enable the autoedge feature on the port.

The autoedge feature allows the port to automatically detect that it is an edge port. If it does not receive any BPDUs in the first three seconds after linkup, enabling, or entering RSTP or MSTP mode, it sets itself to be an edgeport and enters the forwarding state.

Use this command for RSTP or MSTP.

Use the **no** variant of this command to disable this feature.

Syntax

```
spanning-tree autoedge

no spanning-tree autoedge
```

Default Disabled

Mode Interface Configuration

Example

```
          awplus# configure terminal
  awplus(config)# interface port1.0.3
awplus(config-if)# spanning-tree autoedge
```

Related commands spanning-tree edgeport (RSTP and MSTP)

spanning-tree cisco-interoperability (MSTP)

Use this command to enable/disable Cisco-interoperability for MSTP.

Use this command for MSTP only.

Syntax

```
spanning-tree cisco-interoperability {enable|disable}
```

Parameter	Description
`enable`	Enable Cisco interoperability for MSTP.
`disable`	Disable Cisco interoperability for MSTP.

Default If this command is not used, Cisco interoperability is disabled.

Mode Global Configuration

Usage For compatibility with certain Cisco devices, all devices in the switched LAN running the AlliedWare Plus^{TM} Operating System must have Cisco-interoperability enabled. When the AlliedWare Plus^{TM} Operating System is interoperating with Cisco, the only criteria used to classify a region are the region name and revision level. VLAN to instance mapping is not used to classify regions when interoperating with Cisco.

Examples To enable Cisco interoperability on a Layer 2 switch:

```
        awplus#  configure terminal
awplus(config)#  spanning-tree cisco-interoperability enable
```

To disable Cisco interoperability on a Layer 2 switch:

```
        awplus#  configure terminal
awplus(config)#  spanning-tree cisco-interoperability disable
```

spanning-tree edgeport (RSTP and MSTP)

Use this command to set a port as an edge-port.

Use this command for RSTP or MSTP.

This command has the same effect as the spanning-tree portfast (STP) command, but the configuration displays differently in the output of some show commands.

Use the **no** variant of this command to set a port to its default state (not an edge-port).

Syntax

```
spanning-tree edgeport

no spanning-tree edgeport
```

Default Not an edge port.

Mode Interface Configuration

Usage Use this command on a switch port connected to a LAN that has no other bridges attached. If a BPDU is received on the port that indicates that another bridge is connected to the LAN, then the port is no longer treated as an edge port.

Example

```
          awplus# configure terminal
  awplus(config)# interface port1.0.2
awplus(config-if)# spanning-tree edgeport
```

Related commands spanning-tree autoedge (RSTP and MSTP)

spanning-tree enable

Use this command to enable the specified spanning tree protocol on the device. Note that this must be the spanning tree protocol that is configured on the device by the spanning-tree mode command.

Use the **no** variant of this command to disable the configured spanning tree protocol. This places all ports in the forwarding state.

Syntax

```
spanning-tree {mstp|rstp|stp} enable

no spanning-tree {mstp|rstp|stp} enable
```

Parameter	Description
`mstp`	Enables or disables MSTP.
`rstp`	Enables or disables RSTP.
`stp`	Enables or disables STP.

Default The configured spanning tree mode is enabled by default.

Mode Global Configuration

Usage With no configuration, spanning tree is enabled, and the spanning tree mode is set to RSTP. To change the mode, see spanning-tree mode command on page 19.49.

Examples

```
        awplus# configure terminal
awplus(config)# spanning-tree mstp enable
```

```
        awplus# configure terminal
awplus(config)# no spanning-tree mstp enable
```

Related commands spanning-tree mode

spanning-tree errdisable-timeout enable

Use this command to enable the errdisable-timeout facility, which sets a timeout for ports that are disabled due to the BPDU guard feature.

Use this command for RSTP or MSTP.

Use the **no** variant of this command to disable the errdisable-timeout facility.

Syntax

```
spanning-tree errdisable-timeout enable

no spanning-tree errdisable-timeout enable
```

Default By default, the errdisable-timeout is disabled.

Mode Global Configuration

Usage The BPDU guard feature shuts down the port on receiving a BPDU on a BPDU-guard enabled port. This command associates a timer with the feature such that the port is re-enabled without manual intervention after a set interval. This interval can be configured by the user using the spanning-tree errdisable-timeout interval command.

Example

```
        awplus# configure terminal
awplus(config)# spanning-tree errdisable-timeout enable
```

Related Commands

show spanning-tree
spanning-tree errdisable-timeout interval
spanning-tree portfast bpdu-guard

spanning-tree errdisable-timeout interval

Use this command to specify the time interval after which a port is brought back up when it has been disabled by the BPDU guard feature.

Use this command for RSTP or MSTP.

Syntax

```
spanning-tree errdisable-timeout interval <10-1000000>
```

```
no spanning-tree errdisable-timeout interval
```

Parameter	Description
<10-1000000>	Specify the errdisable-timeout interval in seconds.

Default By default, the port is re-enabled after 300 seconds.

Mode Global Configuration

Example

```
        awplus# configure terminal
awplus(config)# spanning-tree errdisable-timeout interval 34
```

Related Commands

show spanning-tree
spanning-tree errdisable-timeout enable
spanning-tree portfast bpdu-guard

spanning-tree force-version

Use this command in Interface Configuration mode for a switch port interface only to force the protocol version for the switch port. Use this command for RSTP or MSTP only.

Syntax

```
spanning-tree force-version <version>
```

```
no spanning-tree force-version
```

Parameter	Description	
<version>	<0-3> Version identifier.	
	0	Forces the port to operate in STP mode.
	1	Not supported.
	2	Forces the port to operate in RSTP mode. If it receives STP BPDUs, it can automatically revert to STP mode.
	3	Forces the port to operate in MSTP mode (this option is only available if MSTP mode is configured). If it receives RSTP or STP BPDUs, it can automatically revert to RSTP or STP mode.

Default By default, no version is forced for the port. The port is in the spanning tree mode configured for the device, or a lower version if it automatically detects one.

Mode Interface Configuration mode for a switch port interface only.

Examples Set the value to enforce the spanning tree protocol (STP):

```
          awplus# configure terminal
  awplus(config)# interface port1.0.2
awplus(config-if)# spanning-tree force-version 0
```

Set the default protocol version:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.2
awplus(config-if)# no spanning-tree force-version
```

Related Commands show spanning-tree

spanning-tree forward-time

Use this command to set the forward delay value. Use the **no** variant of this command to reset the forward delay value to the default setting of 15 seconds.

The **forward delay** sets the time (in seconds) to control how fast a port changes its spanning tree state when moving towards the forwarding state. If the mode is set to STP, the value determines how long the port stays in each of the listening and learning states which precede the forwarding state. If the mode is set to RSTP or MSTP, this value determines the maximum time taken to transition from discarding to learning and from learning to forwarding.

This value is used only when the switch is acting as the root bridge. Switches not acting as the Root Bridge use a dynamic value for the **forward delay** set by the root bridge. The **forward delay**, **max-age**, and **hello time** parameters are interrelated.

Syntax

```
spanning-tree forward-time <forward-delay>

no spanning-tree forward-time
```

Parameter	Description
<forward-delay	<4-30> The forwarding time delay in seconds.

Default

The default is 15 seconds.

Mode

Global Configuration

Usage

The allowable range for forward-time is 4-30 seconds.

The **forward delay**, **max-age**, and **hello time** parameters should be set according to the following formulae, as specified in IEEE Standard 802.1d:

2 x (forward delay - 1.0 seconds) >= max-age

max-age >= 2 x (hello time + 1.0 seconds)

Example

```
        awplus# configure terminal
awplus(config)# spanning-tree forward-time 6
```

Related Commands

show spanning-tree
spanning-tree forward-time <forward-delay>
spanning-tree hello-time <hello-time>
spanning-tree mode

spanning-tree guard root

Use this command in Interface Configuration mode for a switch port only to enable the Root Guard feature for the switch port. The root guard feature disables reception of superior BPDUs. You can use this command for RSTP, STP or MSTP.

Use the **no** variant of this command to disable the root guard feature for the port.

Syntax

```
spanning-tree guard root

no spanning-tree guard root
```

Mode Interface Configuration mode for a switch port interface only.

Usage The Root Guard feature makes sure that the port on which it is enabled is a designated port. If the Root Guard enabled port receives a superior BPDU, it goes to a Listening state (for STP) or discarding state (for RSTP and MSTP).

Example

```
          awplus# configure terminal
  awplus(config)# interface port1.0.2
awplus(config-if)# spanning-tree guard root
```

spanning-tree hello-time

Use this command to set the hello-time. This sets the time in seconds between the transmission of switch spanning tree configuration information when the switch is the Root Bridge of the spanning tree or is trying to become the Root Bridge.

Use this command for RSTP, STP or MSTP.

Use the **no** variant of this command to restore the default of the hello time.

Syntax

```
spanning-tree hello-time <hello-time>
```

```
no spanning-tree hello-time
```

Parameter	Description
<hello-time>	<1-10> The hello BPDU interval in seconds.

Default Default is 2 seconds.

Mode Global Configuration and Interface Configuration for switch ports.

Usage The allowable range of values is 1-10 seconds.

The **forward delay**, **max-age**, and **hello time** parameters should be set according to the following formulae, as specified in IEEE Standard 802.1d:

2 x (**forward delay** - 1.0 seconds) >= **max-age**

max-age >= 2 x (**hello time** + 1.0 seconds)

Example

```
         awplus#  configure terminal
 awplus(config)#  spanning-tree hello-time 3
```

Related Commands spanning-tree forward-time <forward-delay>
spanning-tree max-age <max-age>
show spanning-tree

spanning-tree link-type

Use this command in Interface Configuration mode for a switch port interface only to enable or disable point-to-point or shared link types on the switch port.

Use this command for RSTP or MSTP only.

Use the **no** variant of this command to return the port to the default link type.

Syntax

```
spanning-tree link-type {point-to-point|shared}
```

```
no spanning-tree link-type
```

Parameter	Description
`shared`	Disable rapid transition.
`point-to-point`	Enable rapid transition.

Default The default link type is point-to-point.

Mode Interface Configuration mode for a switch port interface only.

Usage You may want to set link type to shared if the port is connected to a hub with multiple switches connected to it.

Examples

```
           awplus# configure terminal
   awplus(config)# interface port1.0.3
awplus(config-if)# spanning-tree link-type point-to-point
```

spanning-tree max-age

Use this command to set the max-age. This sets the maximum age, in seconds, that dynamic spanning tree configuration information is stored in the switch before it is discarded.

Use this command for RSTP, STP or MSTP.

Use the **no** variant of this command to restore the default of max-age.

Syntax

```
spanning-tree max-age <max-age>

no spanning-tree max-age
```

Parameter	Description
<max-age>	<6-40> The maximum time, in seconds.

Default The default of spanning-tree max-age is 20 seconds.

Mode Global Configuration

Usage Max-age is the maximum time in seconds for which a message is considered valid.

Configure this value sufficiently high, so that a frame generated by the root bridge can be propagated to the leaf nodes without exceeding the max-age.

The **forward delay**, **max-age**, and **hello time** parameters should be set according to the following formulae, as specified in IEEE Standard 802.1d:

2 x (forward delay - 1.0 seconds) >= max-age

max-age >= 2 x (hello time + 1.0 seconds)

Example

```
        awplus# configure terminal
awplus(config)# spanning-tree max-age 12
```

Related Commands show spanning-tree
spanning-tree forward-time <forward-delay>
spanning-tree hello-time <hello-time>

spanning-tree max-hops (MSTP)

Use this command to specify the maximum allowed hops for a BPDU in an MST region. This parameter is used by all the instances of the MST region.

Use the **no** variant of this command to restore the default.

Use this command for MSTP only.

Syntax

```
spanning-tree max-hops <hop-count>
```

```
no spanning-tree max-hops <hop-count>
```

Parameter	Description
<hop-count>	Specify the maximum hops the BPDU will be valid for in the range <1-40>.

Default The default max-hops in a MST region is 20.

Mode Global Configuration

Usage Specifying the max hops for a BPDU prevents the messages from looping indefinitely in the network. The hop count is decremented by each receiving port. When a switch receives an MST BPDU that has a hop count of zero, it discards the BPDU.

Examples

```
        awplus# configure terminal
awplus(config)# spanning-tree max-hops 25
```

```
        awplus# configure terminal
awplus(config)# no spanning-tree max-hops
```

spanning-tree mode

Use this command to change the spanning tree protocol mode on the switch. The spanning tree protocol mode on the switch can be configured to either STP, RSTP or MSTP.

Syntax

```
spanning-tree mode {stp|rstp|mstp}
```

Default The default spanning tree protocol mode on the switch is RSTP.

Mode Global Configuration

Usage With no configuration, the switch will have spanning tree enabled, and the spanning tree mode will be set to RSTP. Use this command to change the spanning tree protocol mode on the device. MSTP is VLAN aware, but RSTP and STP are not VLAN aware. To enable or disable spanning tree operation, see the spanning-tree enable command on page 19.39.

Examples To change the spanning tree mode from the default of RSTP to MSTP, use the following commands:

```
        awplus# configure terminal
awplus(config)# spanning-tree mode mstp
```

Related commands spanning-tree enable

spanning-tree mst configuration

Use this command to enter the MST Configuration mode to configure the Multiple Spanning-Tree Protocol.

Syntax

```
spanning-tree mst configuration
```

Mode Global Configuration

Examples The following example uses this command to enter MST configuration mode. Note the change in the command prompt.

```
         awplus# configure terminal
 awplus(config)# spanning-tree mst configuration
awplus(config-mst)#
```

spanning-tree mst instance

Use this command in Interface Configuration mode to assign a Multiple Spanning Tree instance (MSTI) to a switch port or channel group.

Note that ports are automatically configured to send and receive spanning-tree information for the associated MSTI when VLANs are assigned to MSTIs using the instance vlan (MSTP) command.

Use the **no** variant of this command in Interface Configuration mode to remove the MSTI from the specified switch port or channel group.

Syntax

```
spanning-tree mst instance <instance-id>

no spanning-tree mst instance <instance-id>
```

Parameter	Description
<instance-id>	<1-63> Specify the MST instance ID. The MST instance must have already been created using the instance vlan (MSTP) command.

Default A port automatically becomes a member of an MSTI when it is assigned to a VLAN.

Mode Interface Configuration mode for a switch port or channel group.

Usage You can disable automatic configuration of member ports of a VLAN to an associated MSTI by using a **no spanning-tree mst instance** command to remove the member port from the MSTI. Use the **spanning-tree mst instance** command to add a VLAN member port back to the MSTI.

Examples

```
          awplus# configure terminal
  awplus(config)# interface port1.0.2
awplus(config-if)# spanning-tree mst instance 3

          awplus# configure terminal
  awplus(config)# interface port1.0.2
awplus(config-if)# no spanning-tree mst instance 3
```

Related Commands

instance vlan (MSTP)
spanning-tree mst instance path-cost
spanning-tree mst instance priority
spanning-tree mst instance restricted-role
spanning-tree mst instance restricted-tcn

spanning-tree mst instance path-cost

Use this command in Interface Configuration mode for a switch port interface only to set the cost of a path associated with a switch port, for the specified MSTI (Multiple Spanning Tree Instance) identifier.

This specifies the switch port's contribution to the cost of a path to the MSTI regional root via that port. This applies when the port is the root port for the MSTI.

Use the **no** variant of this command to restore the default cost value of the path.

Syntax

```
spanning-tree mst instance <instance-id> path-cost <path-cost>
```

```
no spanning-tree mst instance <instance-id> path-cost
```

Parameter	Description
<instance-id>	Specify the MSTI identifier in the range <1-63>.
<path-cost>	Specify the cost of path in the range of <1-200000000>, where a lower path-cost indicates a greater likelihood of the specific interface becoming a root.

Default The default path cost values and the range of recommended path cost values depend on the port speed, as shown in the following table from the IEEE 802.1q-2003 standard.

Port speed	Default path cost	Recommended path cost range
Less than 100 Kb/s	200,000,000	20,000,000-200,000,000
1Mbps	20,000,000	2,000,000-20,000,000
10Mbps	2,000,000	200,000-2,000,000
100 Mbps	200,000	20,000-200,000
1 Gbps	20,000	2,000-20,000
10 Gbps	2,000	200-2, 000
100 Gbps	200	20-200
1Tbps	20	2-200
10 Tbps	2	2-20

Mode Interface Configuration mode for a switch port interface only.

Usage Before you can use this command to set a path-cost in a VLAN configuration, you must explicitly add an MST instance to a port using the `spanning-tree instance` command.

Examples

```
          awplus#  configure terminal
  awplus(config)#  interface port1.0.2
awplus(config-if)#  spanning-tree mst instance 3 path-cost 1000
```

Related Commands instance vlan (MSTP)
spanning-tree mst instance
spanning-tree mst instance priority

spanning-tree mst instance restricted-role
spanning-tree mst instance restricted-tcn

spanning-tree mst instance priority

Use this command in Interface Configuration mode for a switch port interface only to set the port priority for an MST instance (MSTI).

Use the **no** variant of this command to restore the default priority value (128).

Syntax

```
spanning-tree mst instance <instance-id> priority <priority>

no spanning-tree mst instance <instance-id> [priority]
```

Parameter	Description
<instance-id>	Specify the MSTI identifier in the range <1-63>.
<priority>	This must be a multiple of 16 and within the range <0-240>. A lower priority indicates greater likelihood of the port becoming the root port.

Default The default is 128.

Mode Interface Configuration mode for a switch port interface.

Usage This command sets the value of the priority field contained in the port identifier. The MST algorithm uses the port priority when determining the root port for the switch in the MSTI. The port with the lowest value is considered to have the highest priority and will be chosen as root port over a port - equivalent in all other aspects - but with a higher priority value.

Examples

```
          awplus# configure terminal
  awplus(config)# interface port1.0.2
awplus(config-if)# spanning-tree mst instance 3 priority 121
```

Related Commands

instance vlan (MSTP)
spanning-tree priority (port priority)
spanning-tree mst instance
spanning-tree mst instance path-cost
spanning-tree mst instance restricted-role
spanning-tree mst instance restricted-tcn

spanning-tree mst instance restricted-role

Use this command in Interface Configuration mode for a switch port interface only to enable the restricted role for an MSTI (Multiple Spanning Tree Instance) on a switch port. Configuring the restricted role for an MSTI on a switch port prevents the switch port from becoming the root port in a spanning tree topology.

Use the **no** variant of this command to disable the restricted role for an MSTI on a switch port. Removing the restricted role for an MSTI on a switch port allows the switch port to become the root port in a spanning tree topology.

Syntax

```
spanning-tree mst instance <instance-id> restricted-role

no spanning-tree mst instance <instance-id> restricted-role
```

Parameter	Description
<instance-id>	<1-63> Specify the MST instance ID. The MST instance must have already been created using the instance vlan (MSTP) command.

Default The restricted role for an MSTI instance on a switch port is disabled by default.

Mode Interface Configuration mode for a switch port interface only.

Usage The root port is the port providing the best path from the bridge to the root bridge. Use this command to disable a port from becoming a root port. Use the **no** variant of this command to enable a port to become a root port. See Spanning tree operation for root port information.

Examples

```
           awplus#  configure terminal
   awplus(config)#  interface port1.0.2
awplus(config-if)#  spanning-tree mst instance 3
                    restricted-role
```

```
           awplus#  configure terminal
   awplus(config)#  interface port1.0.2
awplus(config-if)#  no spanning-tree mst instance 3
                    restricted-role
```

Related Commands

instance vlan (MSTP)
spanning-tree priority (port priority)
spanning-tree mst instance
spanning-tree mst instance path-cost
spanning-tree mst instance restricted-tcn

spanning-tree mst instance restricted-tcn

Use this command in Interface Configuration mode for a switch port interface only to set the restricted TCN (Topology Change Notification) value to TRUE for the specified MSTI (Multiple Spanning Tree Instance).

Use the **no** variant of this command in Interface Configuration mode to reset the restricted TCN for the specified MSTI to the default value of FALSE.

Syntax

```
spanning-tree mst instance <instance-id> restricted-tcn

no spanning-tree mst instance <instance-id> restricted-tcn
```

Parameter	Description
`<instance-id>`	`<1-63>` Specify the MST instance ID. The MST instance must have already been created using the instance vlan (MSTP) command.

Default The default value for restricted TCNs is FALSE, as reset with the **no** variant of this command.

Mode Interface Configuration mode for a switch port interface only.

Usage A Topology Change Notification (TCN) is a simple Bridge Protocol Data Unit (BPDU) that a bridge sends out to its root port to signal a topology change. You can configure restricted TCN between TRUE and FALSE values with this command and the **no** variant of this command.

If you configure restricted TCN to TRUE with this command then this stops the switch port from propagating received topology change notifications and topology changes to other switch ports.

If you configure restricted TCN to FALSE with the **no** variant of this command then this enables the switch port to propagate received topology change notifications and topology changes to other switch ports.

Examples

```
          awplus# configure terminal
  awplus(config)# interface port1.0.2
awplus(config-if)# spanning-tree mst instance 3 restricted-tcn
```

```
          awplus# configure terminal
  awplus(config)# interface port1.0.2
awplus(config-if)# no spanning-tree mst instance 3
                   restricted-tcn
```

Related Commands

instance vlan (MSTP)
spanning-tree priority (port priority)
spanning-tree mst instance
spanning-tree mst instance path-cost
spanning-tree mst instance restricted-role

spanning-tree path-cost

Use this command in Interface Configuration mode for a switch port interface only to set the cost of a path for the specified port. This value then combines with others along the path to the root bridge in order to determine the total cost path value from the particular port, to the root bridge. The lower the numeric value, the higher the priority of the path. This applies when the port is the root port.

Use this command for RSTP, STP or MSTP. When MSTP mode is configured, this will apply to the port's path cost for the CIST.

Syntax

```
spanning-tree path-cost <pathcost>

no spanning-tree path-cost
```

Parameter	Description
<pathcost>	<1-200000000> The cost to be assigned to the port.

Default The default path cost values and the range of recommended path cost values depend on the port speed, as shown in the following table from the IEEE 802.1q-2003 and IEEE 802.1d-2004 standards.

Port speed	Default path cost	Recommended path cost range
Less than 100 Kb/s	200,000,000	20,000,000-200,000,000
1Mbps	20,000,000	2,000,000-20,000,000
10Mbps	2,000,000	200,000-2,000,000
100 Mbps	200,000	20,000-200,000
1 Gbps	20,000	2,000-20,000
10 Gbps	2,000	200-2, 000
100 Gbps	200	20-200
1Tbps	20	2-200
10 Tbps	2	2-20

Mode Interface Configuration mode for switch port interface only.

Example

```
          awplus# configure terminal
  awplus(config)# interface port1.0.2
awplus(config-if)# spanning-tree path-cost 123
```

spanning-tree portfast (STP)

Use this command in Interface Configuration mode for a switch port interface only to set a port as an edge-port. The portfast feature enables a port to rapidly move to the forwarding state, without having first to pass through the intermediate spanning tree states. This command has the same effect as the spanning-tree edgeport (RSTP and MSTP) command, but the configuration displays differently in the output of some show commands.

Note You can run either of two additional parameters with this command. To simplify the syntax these are documented as separate commands. See the following additional portfast commands:

- spanning-tree portfast bpdu-filter command on page 19.58
- spanning-tree portfast bpdu-guard command on page 19.60.

You can obtain the same effect by running the spanning-tree edgeport (RSTP and MSTP) command. However, the configuration output may display differently in some show commands.

Use the **no** variant of this command to set a port to its default state (not an edge-port).

Syntax

```
spanning-tree portfast

no spanning-tree portfast
```

Default Not an edge port.

Mode Interface Configuration mode for a switch port interface only.

Usage Portfast makes a port move from a blocking state to a forwarding state, bypassing both listening and learning states. The portfast feature is meant to be used for ports connected to end-user devices not switches. Enabling portfast on ports that are connected to a workstation or server allows devices to connect to the network without waiting for spanning-tree to converge. For example, you may need hosts to receive a DHCP address quickly and waiting for STP to converge would cause the DHCP request to time out. Ensure you do not use portfast on any ports connected to another switch to avoid creating a spanning-tree loop on the network.

Use this command on a switch port that connects to a LAN with no other bridges attached. An edge port should never receive BPDUs. Therefore if an edge port receives a BPDU, the portfast feature takes one of three actions.

- Cease to act as an edge port and pass BPDUs as a member of a spanning tree network (spanning-tree portfast (STP) command disabled).
- Filter out the BPDUs and pass only the data and continue to act as a edge port (spanning-tree portfast bpdu-filter command enabled)
- Block the port to all BPDUs and data (spanning-tree portfast bpdu-guard command enabled).

Example

```
        awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# spanning-tree portfast
```

Related Commands spanning-tree edgeport (RSTP and MSTP)
show spanning-tree
spanning-tree portfast bpdu-filter
spanning-tree portfast bpdu-guard

spanning-tree portfast bpdu-filter

This command sets the portfast bpdu-filter feature and applies a filter to any BPDUs received. Enabling this feature ensures that portfast configured ports will not transmit any BPDUs and will ignore (filter out) any BPDUs received. BPDU Filter is not enabled on a port by default.

Using the **no** variant of this command to turn off the bpdu-filter, but retain the port's status as a portfast enabled port. If the port then receives a BPDU it will change its role from an **edge-port** to a **non edge-port**.

Syntax (Global Configuration)

```
spanning-tree portfast bpdu-filter

no spanning-tree portfast bpdu-filter
```

Syntax (Interface Configuration)

```
spanning-tree portfast bpdu-filter {default|disable|enable}

no spanning-tree portfast bpdu-filter
```

Parameter	Description	
`portfast`	A port that behaves as an edge-port. Note that an edge-port should never receive BPDUs. If a port does receive a BPDU then it will filter any received.	
`bpdu-filter`	A portfast port that has bpdu-filter enabled will not transmit any BPDUs and will ignore any BPDUs received. This port type has one of the following parameters (in Interface Configuration mode):	
	`default`	Takes the setting that has been configured for the whole switch, i.e. the setting made from the Global configuration mode.
	`disable`	Turns off BPDU filter.
	`enable`	Turns on BPDU filter.

Default BPDU Filter is not enabled on any ports by default.

Mode Global Configuration and Interface Configuration

Usage This command filters the BPDUs and passes only data to continue to act as an edge port. Using this command in Global Configuration mode applies the portfast bpdu-filter feature to all ports on the switch. Using it in Interface mode applies the portfast feature to a specific port, or range of ports.The command will operate in both RSTP and MSTP networks.

Use the show spanning-tree command to display status of the bpdu-filter parameter for the switch ports.

Example

```
          awplus# configure terminal
  awplus(config)# spanning-tree portfast bpdu-filter

          awplus# configure terminal
  awplus(config)# interface port1.0.2
awplus(config-if)# spanning-tree portfast bpdu-filter enable
```

Related Commands

spanning-tree edgeport (RSTP and MSTP)
show spanning-tree
spanning-tree portfast (STP)
spanning-tree portfast bpdu-guard

spanning-tree portfast bpdu-guard

This command sets the portfast feature and applies a BPDU guard to the port. A port with the portfast bpdu-guard feature enabled will block all traffic (BPDUs and user data), if it starts receiving BPDUs.

Use this command in Global Configuration mode to set the portfast feature and apply BPDU guard to all ports on the switch. Use this command in Interface mode to for an individual interface or a range of interfaces specified. BPDU Guard is not enabled on a port by default.

Use the **no** variant of this command to disable the BPDU Guard feature on a switch in Global Configuration mode or to disable the BPDU Guard feature on a port in Interface mode.

Syntax (Global Configuration)

```
spanning-tree portfast bpdu-guard

no spanning-tree portfast bpdu-guard
```

Syntax (Interface Configuration)

```
spanning-tree portfast bpdu-guard {default|disable|enable}

no spanning-tree portfast bpdu-guard
```

Parameter	Description	
`portfast`	A port that behaves as an edge-port. Note that an edge port should never receive BPDUs. If a port does receive a BPDU then it will cease to act as an edge port.	
`bpdu-guard`	A portfast port that has bpdu-guard turned on will enter the STP blocking state if it receives a BPDU. This port type has one of the following parameters (in Interface Configuration mode):	
	`default`	Takes the setting that has been configured for the whole switch, i.e. the setting made from the Global configuration mode.
	`disable`	Turns off BPDU guard.
	`enable`	Turns on BPDU guard and will also set the port as an edge port.

Default BPDU Guard is not enabled on any ports by default.

Mode Global Configuration or Interface Configuration

Usage This command blocks the port(s) to all BPDUs and data when enabled. BPDU Guard is a port-security feature that changes how a portfast-enabled port behaves if it receives a BPDU. When **bpdu-guard** is set, then the port shuts down if it receives a BPDU. It does not process the BPDU as it is considered suspicious. When **bpdu-guard** is not set, then the port will negotiate spanning-tree with the device sending the BPDUs. By default, bpdu-guard is not enabled on a port. If a port with portfast enabled receives a BPDU, the port will be moved to the disabled state. This stops the port being connected to another port that is configured with portfast, so guards against spanning-tree loops forming on the network.

You can configure a port disabled by the bpdu-guard to re-enable itself after a specific time interval. This interval is set with the spanning-tree errdisable-timeout interval command on page 19.41. If you do not use the **errdisable-timeout** feature, then you will need to manually re-enable the port by using the **no shutdown** command.

Use the show spanning-tree command on page 19.15 to display the switch and port configurations for the BPDU Guard feature. It shows both the administratively configured and currently running values of bpdu-guard.

Example

```
         awplus# configure terminal
 awplus(config)# spanning-tree portfast bpdu-guard
```

```
            awplus# configure terminal
    awplus(config)# interface port1.0.2
 awplus(config-if)# spanning-tree portfast bpdu-guard enable
```

Related Commands

spanning-tree edgeport (RSTP and MSTP)
show spanning-tree
spanning-tree portfast (STP)
spanning-tree portfast bpdu-guard

spanning-tree priority (bridge priority)

Use this command to set the bridge priority for the switch. A lower priority value indicates a greater likelihood of the switch becoming the root bridge.

Use this command for RSTP, STP or MSTP. When MSTP mode is configured, this will apply to the CIST.

Use the **no** variant of this command to reset it to the default.

Syntax

```
spanning-tree priority <priority>
```

```
no spanning-tree priority
```

Parameter	Description
<priority>	<0-61440> The bridge priority, which will be rounded to a multiple of 4096.

Default The default priority is 32678.

Mode Global Configuration

Usage To force a particular switch to become the root bridge use a lower value than other switches in the spanning tree.

Example

```
        awplus# configure terminal
awplus(config)# spanning-tree priority 4096
```

Related Commands spanning-tree mst instance priority
show spanning-tree

spanning-tree priority (port priority)

Use this command in Interface Configuration mode for a switch port interface only to set the port priority for port. A lower priority value indicates a greater likelihood of the port becoming part of the active topology.

Use this command for RSTP, STP, or MSTP. When the device is in MSTP mode, this will apply to the CIST.

Use the **no** variant of this command to reset it to the default.

Syntax

```
spanning-tree priority <priority>
```

```
no spanning-tree priority
```

Parameter	Description
<priority>	<0-240>, in increments of 16. The port priority, which will be rounded down to a multiple of 16.

Default The default priority is 128.

Mode Interface Configuration mode for a switch port interface only.

Usage To force a port to be part of the active topology (for instance, become the root port or a designated port) use a lower value than other ports on the device. (This behavior is subject to network topology, and more significant factors, such as bridge ID.)

Example

```
        awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# spanning-tree priority 16
```

Related Commands spanning-tree mst instance priority
spanning-tree priority (bridge priority)
show spanning-tree

spanning-tree restricted-role

Use this command in Interface Configuration mode for a switch port interface only to restrict the port from becoming a root port.

Use the **no** variant of this command to disable the restricted role functionality.

Syntax

```
spanning-tree restricted-role
no spanning-tree restricted-role
```

Default The restricted role is disabled.

Mode Interface Configuration mode for a switch port interface only.

Example

```
          awplus# configure terminal
   awplus(config)# interface port1.0.2
awplus(config-if)# spanning-tree restricted-role
```

spanning-tree restricted-tcn

Use this command in Interface Configuration mode for a switch port interface only to prevent TCN (Topology Change Notification) BPDUs (Bridge Protocol Data Units) from being sent on a port. If this command is enabled, after a topology change a bridge is prevented from sending a TCN to its designated bridge.

Use the **no** variant of this command to disable the restricted TCN functionality.

Syntax

```
spanning-tree restricted-tcn

no spanning-tree restricted-tcn
```

Default The restricted TCN is disabled.

Mode Interface Configuration mode for a switch port interface only.

Example

```
          awplus# configure terminal
  awplus(config)# interface port1.0.2
awplus(config-if)# spanning-tree restricted-tcn
```

spanning-tree transmit-holdcount

Use this command to set the maximum number of BPDU transmissions that are held back.

Use the **no** variant of this command to restore the default transmit hold-count value.

Syntax

```
spanning-tree transmit-holdcount <1-10>

no spanning-tree transmit-holdcount <1-10>
```

Parameter	Description
<1-10>	Transmit hold-count value.

Default Transmit hold-count default is 3.

Mode Global Configuration

Example

```
        awplus# configure terminal
awplus(config)# spanning-tree transmit-holdcount 5
```

undebug mstp

This command applies the functionality of the no debug mstp (RSTP and STP) command.

Chapter 20: Link Aggregation Introduction and Configuration

Introduction

This chapter contains a sample Link Aggregation Control Protocol (LACP), or dynamic channel group, configuration and a sample static channel group configuration.

To see details about the commands used to configure dynamic (LACP) and static Link aggregation, see Chapter 21, Link Aggregation Commands.

For a brief overview of static and dynamic link aggregation (LACP), see Static and Dynamic (LACP) Link Aggregation.

Link Aggregation Control Protocol (LACP)

LACP is based on the IEEE Standard 802.3ad. It allows bundling of several physical ports to form a single logical channel providing enhanced performance and resiliency. The aggregated channel is viewed as a single link by each switch. Spanning tree also views the channel as one interface and not as multiple interfaces. When there is a failure in one physical port, the other ports stay up and there is no disruption.

This device supports the aggregation of a maximum of eight physical ports into a single channel group.

Note AlliedWare Plus^{TM} supports IEEE 802.3ad link aggregation and uses the Link Aggregation Control Protocol (LACP). LACP does not interoperate with devices that use Port Aggregation Protocol (PAgP).

Note Link aggregation does not necessarily achieve exact load balancing across the links. The load sharing algorithm is designed to ensure that any given data flow always goes down the same link. It also aims to spread data flows across the links as evenly as possible.

Link aggregation hashes the source and destination MAC address, IP address and UDP/TCP ports to select a link on which to send a packet. So packet flow between a pair of hosts always takes the same link inside the Link Aggregation Group (LAG). The net effect is that the bandwidth for a given packet stream is restricted to the speed of one link in the LAG.

For example, for a 2 Gbps LAG that is a combination of two 1 Gbps ports, one flow of traffic can only ever reach a maximum throughput of 1 Gbps. However, the hashing algorithm should spread the flows across the links so that when many flows are operating, the full 2 Gbps can be utilized.

For information about load balancing see the platform load-balancing command.

LACP operates where systems are connected over multiple communications links. Once LACP has been initially configured and enabled, it automatically aggregates the ports that have been assigned to a channel group, if possible. LACP continues to monitor these groups and dynamically adds or removes links to them as network changes occur.

LACP achieves this by determining:

- which ports are under LACP control (channel-group command on page 21.4)
- whether each port is in LACP active or LACP passive mode (channel-group command on page 21.4)
- which system has the highest LACP priority (lacp system-priority command on page 21.8)
- the LACP priority of ports (lacp port-priority command on page 21.7)
- whether the LACP timeout is short or long_(lacp timeout command on page 21.9)

Channel group identification

In order to identify particular channel groups, each group is assigned a link aggregation identifier called a **lag ID**. The lag ID comprises the following components for both the local system (called the Actor) followed by their equivalent components for the remote system (called the Partner):

- system identifier - the MAC address of the system
- port key - An identifier - created by the LACP software
- port priority - set by the lacp port-priority command on page 21.7
- port number - determined by the device connection

The lag ID can be displayed for each aggregated link by entering the show etherchannel command on page 21.12.

Static and Dynamic (LACP) Link Aggregation

Channels, either static or dynamic LACP, increase reliability by distributing the data path over more than one physical link. Channels must be configured on both ends of a link or network loops may result. Ports in a channel group need not be contiguous. A mirror port cannot be a member of either a static or a dynamic channel group.

Aggregation criteria

For individual links to be aggregated into a channel group they must:

- originate on the same device or stack
- terminate on the same device or stack
- be members of the same VLANs (**vlan** command on page 17.31)
- have the same data rate (**speed** command on page 15.43)
- share the same admin port key (assigned by using the **channel-group** command on page 21.4 command)
- be operating in full duplex mode (**duplex** command on page 15.9)

The hardware must also be capable and have the capacity to handle the number of links to be aggregated.

Static Channel Groups

A static channel group, also known as a static aggregator, enables a number of ports to be manually configured to form a single logical connection of higher bandwidth. By using static channel groups you increase channel reliability by distributing the data path over more than one physical link.

Dynamic (LACP) Channel Groups

A LACP channel group, also known as an etherchannel, a LACP aggregator, or a dynamic channel group, enables a number of ports to be dynamically combined to form a single higher bandwidth logical connection.

For LACP configuration examples see Configuring an LACP Channel Group, Configuring a Static Channel Group, and Configuring a Dynamic Channel Group sections in this chapter.

For details of LACP channel group commands, see Chapter 21, Link Aggregation Commands.

Configuring an LACP Channel Group

The following example shows how to configure three links between two Allied Telesis managed Layer 3 Switches. The three links are assigned the same administrative key (1), so that they aggregate to form a single channel (1). They are viewed by the STP as one interface.

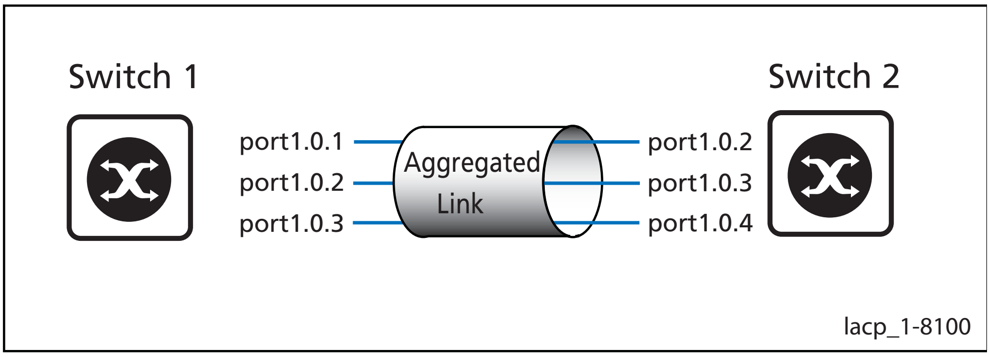


Table 20-1: Switch 1 configuration

`awplus#`
`configure terminal`
	Enter the Global Configuration mode.

`awplus(config)#`
`lacp system-priority 20000`
	Set the system priority of this switch. This priority is used to determine which switch in the system is responsible for resolving conflicts in the choice of aggregation groups. A lower numerical value has a higher priority. Switch 1 has a higher priority than Switch 2 in this configuration.

`awplus(config)#`
`interface port1.0.1`
	Enter the Interface Configuration mode to configure port `1.0.1`.

`awplus(config-if)#`
`channel-group 1 mode active`
	Add this interface to a channel group `1` and enable link aggregation so that it may be selected for aggregation by the local system.

`awplus(config-if)#`
`exit`
	Exit the Interface Configuration mode and return to the Global Configure mode.

`awplus(config)#`
`interface port1.0.2`
	Enter the Interface Configuration mode to configure port `1.0.2`.

`awplus(config-if)#`
`channel-group 1 mode active`
	Add this interface to a channel group `1` and enable link aggregation so that it may be selected for aggregation by the local system.

`awplus(config-if)#`
`exit`
	Exit the Interface Configuration mode and return to the Global Configure mode.

Table 20-1: Switch 1 configuration (cont.)

awplus(config)# interface port1.0.3	Enter the Interface Configuration mode to configure port 1.0.3.
awplus(config-if)# channel-group 1 mode active	Add this interface to a channel group 1 and enable link aggregation so that it may be selected for aggregation by the local system.
awplus(config-if)# interface po1	Select the dynamic aggregator logical interface created for channel-group 1 named po1.

Table 20-2: Switch 2 configuration

awplus# configure terminal	Enter the Global Configuration mode.
awplus(config)# lacp system-priority 3000	Set the system priority of this switch. This priority is used to determine which switch in the system is responsible for resolving conflicts in the choice of aggregation groups. A lower numerical value has a higher priority. Switch 2 has a lower priority than Switch 1 in this configuration.
awplus(config)# interface port1.0.2	Enter the Interface Configuration mode to configure port 1.0.2.
awplus(config-if)# channel-group 1 mode active	Add this interface to a channel group 1 and enable link aggregation so that it may be selected for aggregation by the local system.
awplus(config-if)# exit	Exit the Interface mode and return to the Configure mode.
awplus(config)# interface port1.0.3	Enter the Interface mode to configure port 1.0.3.
awplus(config-if)# channel-group 1 mode active	Add this interface to a channel group 1 and enable link aggregation so that it may be selected for aggregation by the local system.
awplus(config-if)# exit	Exit the Interface Configuration mode and return to the Global Configuration mode.

Table 20-2: Switch 2 configuration

`awplus(config)#`
`interface port1.0.4`
	Enter the Interface Configuration mode to configure port `1.0.4`.

`awplus(config-if)#`
`channel-group 1 mode active`
	Add this interface to a channel group `1` and enable link aggregation so that it may be selected for aggregation by the local system.

`awplus(config-if)#`
`interface po1`
	Select the dynamic aggregator logical interface created for channel-group 1 named `po1`.

Commands Used

lacp system-priority
channel-group

Validation Commands

show lacp sys-id
show port etherchannel
show etherchannel
show etherchannel detail

Configuring a Static Channel Group

For details of LACP channel group commands, see Chapter 21, Link Aggregation Commands.

The following example creates a static channel group and adds switch ports `1.0.1` and `1.0.2`.

awplus# `configure terminal`	Enter the Global Configuration mode.
awplus(config)# `interface port1.0.1`	Enter the Interface Configuration mode to configure port `1.0.1`.
awplus(config-if)# `static-channel-group 2`	Add port `1.0.1` to static-channel-group 2.
awplus(config-if)# `exit`	Exit the Interface Configuration mode and return to the Global Configuration mode.
awplus(config)# `interface port1.0.2`	Enter the Interface Configuration mode to configure port `1.0.2`.
awplus(config-if)# `static-channel-group 2`	Add port `1.0.2` to static-channel-group 2.
awplus(config-if)# `interface sa2`	Select the static aggregator logical interface created for static-channel-group 2 named `sa2`.

Commands Used static-channel-group

Validation Commands show static-channel-group

Configuring a Dynamic Channel Group

For details of LACP channel group commands, see Chapter 21, Link Aggregation Commands.

The following example creates LACP channel group 2 and enables link aggregation on switch ports `1.0.1` and `1.0.2` within this channel group. Note that all aggregated ports must belong to the same VLAN.

awplus# `configure terminal`	Enter Global Configuration mode.
awplus(config)# `interface port1.0.1-port1.0.2`	Enter the Interface Configuration mode for the switch ports to aggregate into the channel group.
awplus(config-if)# `channel-group 2 mode active`	Assign the switch ports to channel group 2 in active mode. This creates the channel group.
awplus(config-if)# `interface po2`	Select the dynamic aggregator logical interface created for channel-group 2 named `po2`.

Commands Used channel-group

Validation Commands show static-channel-group

Chapter 21: Link Aggregation Commands

Introduction

This chapter provides an alphabetical reference of commands used to configure a static channel group (static aggregator) and dynamic channel group (LACP channel group, etherchannel or LACP aggregator). Link aggregation is also sometimes referred to as channelling.

Note AlliedWare Plus^{TM} supports IEEE 802.3ad link aggregation and uses the Link Aggregation Control Protocol (LACP). LACP does not interoperate with devices that use Port Aggregation Protocol (PAgP).

Note LACP does not perform load balancing. The LACP algorithm is based on the packet flow. Link aggregation (LAG) hashes the source and destination MAC address, IP address and UDP/TCP ports to select a port on which to send a packet. So packet flow between a pair of hosts always takes the same port inside the LAG. The net effect is that the bandwidth for one packet stream is restricted to the speed of one link in the LAG. For example, for a 2 Gbps LAG that is a combination of two 1 Gbps ports, one flow of traffic can only ever reach a maximum throughput of 1 Gbps.

For information about load balancing see the platform load-balancing command on page 15.23 command.

For a description of static and dynamic link aggregation (LACP), see “Static and Dynamic (LACP) Link Aggregation” on page 20.4. For an LACP configuration example, see Chapter 20, Link Aggregation Introduction and Configuration.

Command List

clear lacp counters

Use this command to clear all counters of all present LACP aggregators (channel groups) or a given LACP aggregator.

Syntax `clear lacp [<1-32>] counters`

Parameter	Description
<1-32>	Channel-group number.

Mode Privileged Exec

Example

```
awplus# clear lacp 2 counters
```

channel-group

Use this command to add the switch port to a dynamic channel group specified by the dynamic channel group number, and set its mode. This command enables LACP link aggregation on the switch port, so that it may be selected for aggregation by the local system. Dynamic channel groups are also known as LACP channel groups, LACP aggregators or etherchannels.

You can create up to 32 dynamic (LACP) channel groups (and up to 96 static channel groups).

Use the **no** variant of this command to turn off link aggregation on the switch port.

Syntax

```
channel-group <dynamic-channel-group-number> mode {active|passive}
```

```
no channel-group
```

Parameter	Description
<dynamic-channel-group-number>	<1-32> Specify a dynamic channel group number for an LACP link. A maximum of 32 combined dynamic and static channel groups is supported with the base license. The optional LAG-128 feature licence extends the maximum number of combined dynamic and static channel groups supported to 128 with up to 32 dynamic channel groups and up to 96 static channel groups.
active	Enables initiation of LACP negotiation on a port. The port will transmit LACP dialogue messages whether or not it receives them from the partner system.
passive	Disables initiation of LACP negotiation on a port. The port will only transmit LACP dialogue messages if the partner systems is transmitting them, i.e. the partner is in the active mode.

Mode Interface Configuration

Usage All the switch ports in a channel-group must belong to the same VLANs, have the same tagging status, and can only be operated on as a group. All switch ports within a channel group must have the same port speed and be in full duplex mode.

Once the LACP channel group has been created, it is treated as a switch port, and can be referred to in most other commands that apply to switch ports.

To refer to an LACP channel group in other LACP commands, use the channel group number. To specify an LACP channel group (LACP aggregator) in other commands, prefix the channel group number with **po**. For example, 'po4' refers to the LACP channel group with channel group number 4.

For more on LACP, see "Dynamic (LACP) Channel Groups" on page 20.4 and Chapter 20, Link Aggregation Introduction and Configuration.

Examples To add switch port1.0.10 to a newly created LACP channel group 4 use the commands below:

```
           awplus# configure terminal
   awplus(config)# interface port1.0.10
awplus(config-if)# channel-group 4 mode active
```

To remove switch `port1.0.8` from any created LACP channel groups use the command below:

```
        awplus#  configure terminal
awplus(config)#  interface port1.0.8
awplus(config-if)#  no channel-group
```

To reference the pre-defined LACP channel group 2 as an interface apply commands as below:

```
        awplus#  configure terminal
awplus(config)#  interface port1.0.8
awplus(config-if)#  channel-group 2 mode active
awplus(config-if)#  exit
awplus(config)#  interface port.1.0.10
awplus(config-if)#  channel-group 2 mode active
awplus(config-if)#  exit
awplus(config)#  interface po2
awplus(config-if)#
```

Related Commands

show etherchannel
show etherchannel detail
show etherchannel summary
show port etherchannel

debug lacp

Use this command to enable all LACP troubleshooting functions.

Use the **no** variant of this command to disable this function.

Syntax

```
debug lacp {all|cli|event|ha|packet|sync|timer[detail]}
```

```
no debug lacp {all|cli|event|ha|packet|sync|timer[detail]}
```

Parameter	Description
`all`	Turn on all debugging for LACP.
`cli`	Specifies debugging for CLI messages. Echoes commands to the console.
`event`	Specifies debugging for LACP events. Echoes events to the console.
`ha`	Specifies debugging for HA (High Availability) events. Echoes High Availability events to the console.
`packet`	Specifies debugging for LACP packets. Echoes packet contents to the console.
`sync`	Specified debugging for LACP synchronization. Echoes synchronization to the console.
`timer`	Specifies debugging for LACP timer. Echoes timer expiry to the console.
`detail`	Optional parameter for LACP timer-detail. Echoes timer start/stop details to the console.

Mode Privileged Exec and Global Configuration

Examples

```
awplus# debug lacp timer detail
```

```
awplus# debug lacp all
```

Related Commands show debugging lacp
undebug lacp

lacp port-priority

Use this command to set the priority of a switch port. Ports are selected for aggregation based on their priority, with the higher priority (numerically lower) ports selected first.

Use the **no** variant of this command to reset the priority of port to the default.

Syntax

```
lacp port-priority <1-65535>

no lacp port-priority
```

Parameter	Description
<1-65535>	Specify the LACP port priority.

Default The default is 32768.

Mode Interface Configuration

Example

```
awplus# configure terminal
awplus(config)# interface port1.0.5
awplus(config-if)# lacp port-priority 34
```

lacp system-priority

Use this command to set the system priority of a local system. This is used in determining the system responsible for resolving conflicts in the choice of aggregation groups.

Use the **no** variant of this command to reset the system priority of the local system to the default.

Syntax

```
lacp system-priority <1-65535>

no lacp system-priority
```

Parameter	Description
<1-65535>	LACP system priority. Lower numerical values have higher priorities.

Default The default is 32768.

Mode Global Configuration

Example

```
        awplus# configure terminal
awplus(config)# lacp system-priority 6700
```

lacp timeout

Use this command to set the short or long timeout on a port. Ports will time out of the aggregation if three consecutive updates are lost.

Syntax `lacp timeout {short|long}`

Parameter	Description
`timeout`	Number of seconds before invalidating a received LACP data unit (DU).
`short`	LACP short timeout. The **short** timeout value is **1** second.
`long`	LACP long timeout. The **long** timeout value is **30** seconds.

Default The default is **long** timeout (30 seconds).

Mode Interface Configuration

Usage This command enables the switch to indicate the rate at which it expects to receive LACPDUs from its neighbor.

If the timeout is set to **long**, then the switch expects to receive an update every **30** seconds, and this will time a port out of the aggregation if no updates are seen for 90 seconds (i.e. 3 consecutive updates are lost).

If the timeout is set to **short**, then the switch expects to receive an update every second, and this will time a port a port out of the aggregation if no updates are seen for 3 seconds (i.e. 3 consecutive updates are lost).

The switch indicates its preference by means of the '`Timeout`' field in the '`Actor`' section of its LACPDUs. If the '`Timeout`' field is set to 1, then the switch has set the **short** timeout. If the '`Timeout`' field is set to 0, then the switch has set the **long** timeout.

Setting the **short** timeout enables the switch to be more responsive to communication failure on a link, and does not add too much processing overhead to the switch (1 packet per second).

Note It is not possible to configure the rate that the switch sends LACPDUs; the switch must send at the rate which the neighbor indicates it expects to receive LACPDUs.

Examples The following commands set the LACP long timeout period for 30 seconds on `port1.0.2`.

```
          awplus#  configure terminal
  awplus(config)#  interface port1.0.2
awplus(config-if)# lacp timeout long
```

The following commands set the LACP short timeout for 1 second on `port1.0.2`.

```
          awplus#  configure terminal
  awplus(config)#  interface port1.0.2
awplus(config-if)# lacp timeout short
```

show debugging lacp

Use this command to display the LACP debugging option set.

For information on output options, see “Controlling “show” Command Output” on page 1.41.

Syntax `show debugging lacp`

Mode User Exec and Privileged Exec

Example

```
awplus# show debugging lacp
```

Output Figure 21-1: Example output from the **show debugging lacp** command

```
LACP debugging status:
 LACP timer debugging is on
 LACP timer-detail debugging is on
 LACP cli debugging is on
 LACP packet debugging is on
 LACP event debugging is on
 LACP sync debugging is on
```

Related Commands debug lacp

show diagnostic channel-group

This command displays dynamic and static channel group interface status information. The output of this command is useful for Allied Telesis authorized service personnel for diagnostic purposes.

For information on output options, see “Controlling “show” Command Output” on page 1.41.

Syntax

```
show diagnostic channel-group
```

Mode User Exec and Privileged Exec

Example

```
awplus# show diagnostic channel-group
```

Output Figure 21-2: Example output from the **show diagnostic channel-group** command

```
awplus#show diagnostic channel-group

Channel Group Info based on NSM:
Note: Pos - position in hardware table
-------------------------------------------------------------
Dev  Interface  IfIndex  Member port  IfIndex  Active  Pos
-------------------------------------------------------------
     sa3        4503     port1.0.15   5015     No
     sa3        4503     port1.0.18   5018     No
     po1        4601     port1.0.7    5007     No
     po1        4601     port1.0.8    5008     No
     po1        4601     port1.0.9    5009     No


Channel Group Info based on HSL:
Note: Pos - position in hardware table
-------------------------------------------------------------
Dev  Interface  IfIndex  Member port  IfIndex  Active  Pos
-------------------------------------------------------------
     sa3        4503                           N/a
     po1        4601                           N/a


Channel Group Info based on IPIFWD:
Note: Pos - position in hardware table
-------------------------------------------------------------
Dev  Interface  IfIndex  Member port  IfIndex  Active  Pos
-------------------------------------------------------------
     sa3        4503                           N/a
     po1        4601                           N/a


Channel Group Info based on HW:
Note: Pos - position in hardware table
      Only entries from first device are displayed.
-------------------------------------------------------------
Dev  Interface  IfIndex  Member port  IfIndex  Active  Pos
-------------------------------------------------------------
     sa3        4503                           N/a
     po1        4601                           N/a


No error found
```

Related Commands show tech-support

show etherchannel

Use this command to display information about a LACP channel specified by the channel group number.

The command output also shows the thrash limiting status. If thrash limiting is detected and the **thrash limiting** parameter of the thrash-limiting command on page 15.50 is set to **vlan disable**, the output will also show the VLANs on which thrashing is detected.

For information on output options, see “Controlling “show” Command Output” on page 1.41.

Syntax

```
show etherchannel [<1-32>]
```

Parameter	Description
<1-32>	Channel-group number.

Mode User Exec and Privileged Exec

Example

```
awplus# show etherchannel 5
```

Output Figure 21-3: Example output from the **show etherchannel** command

```
% Lacp Aggregator: po1
 Thrash-limiting
 Status Vlan Thrashing Detected, Action vlan-disable 60(s)
 Thrashing Vlans 1 2 3 4 5
% Member:
   port1.0.4
   port1.0.8
```

show etherchannel detail

Use this command to display detailed information about all LACP channels.

For information on output options, see “Controlling “show” Command Output” on page 1.41.

Syntax

```
show etherchannel detail
```

Mode User Exec and Privileged Exec

Example

```
awplus# show etherchannel detail
```

Output Figure 21-4: Example output from the **show etherchannel detail** command

```
% Aggregator po1 (4501)
%  Mac address: 00:00:cd:24:fd:29
%  Admin Key: 0001 - Oper Key 0001
%  Receive link count: 1 - Transmit link count: 0
%  Individual: 0 - Ready: 1
%  Partner LAG: 0x8000,00-00-cd-24-da-a7
%   Link: port1.0.1 (5001) disabled
%   Link: port1.0.2 (5002) sync: 1
% Aggregator po2 (4502)
%  Mac address: 00:00:cd:24:fd:29
%  Admin Key: 0002 - Oper Key 0002
%  Receive link count: 1 - Transmit link count: 0
%  Individual: 0 - Ready: 1
%  Partner LAG: 0x8000,00-00-cd-24-da-a7
%   Link: port1.0.7 (5007) disabled
```

show etherchannel summary

Use this command to display a summary of all LACP channels.

For information on output options, see “Controlling “show” Command Output” on page 1.41.

Syntax

```
show etherchannel summary
```

Mode

User Exec and Privileged Exec

Example

```
awplus# show etherchannel summary
```

Output

Figure 21-5: Example output from the **show etherchannel summary** command

```
% Aggregator po1
%  Admin Key: 0001 - Oper Key 0001
%   Link: port1.0.1 (5001) disabled
%   Link: port1.0.2 (5002) sync: 1
% Aggregator po2
%  Admin Key: 0002 - Oper Key 0002
%   Link: port1.0.7 (5007) disabled
```

show lacp-counter

Use this command to display the packet traffic on all ports of all present LACP aggregators, or a given LACP aggregator.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax `show lacp-counter [<1-32>]`

Parameter	Description
<1-32>	Channel-group number.

Mode User Exec and Privileged Exec

Example

```
awplus# show lacp-counter 2
```

Output Figure 21-6: Example output from the **show lacp-counter** command

```
% Traffic statistics
Port          LACPDUs         Marker         Pckt err
           Sent    Recv    Sent    Recv    Sent    Recv
% Aggregator po4 (4604)
port1.0.2   0       0       0       0       0       0
```

show lacp sys-id

Use this command to display the LACP system ID and priority.

For information on output options, see “Controlling “show” Command Output” on page 1.41.

Syntax

```
show lacp sys-id
```

Mode User Exec and Privileged Exec

Example

```
awplus# show lacp sys-id
```

Output Figure 21-7: Example output from the **show lacp sys-id** command

```
% System Priority: 0x8000 (32768)
% MAC Address: 00-00-cd-24-fd-29
```

show port etherchannel

Use this command to show LACP details of the switch port specified.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax `show port etherchannel <port>`

Parameter	Description
<port>	Name of the switch port to display LACP information about.

Mode User Exec and Privileged Exec

Example

```
awplus# show port etherchannel port1.0.1
```

Output Figure 21-8: Example output from the **show port etherchannel** command

```
% Link: port1.0.1 (5001)
% Aggregator: po1 (4501)
% Receive machine state: Current
% Periodic Transmission machine state: Fast periodic
% Mux machine state: Collecting/Distributing
% Actor Information:                     Partner Information:
%   Selected ................ Selected     Partner Sys Priority ............ 0
%   Physical Admin Key ............. 1     Partner System .. 00-00-00-00-00-00
%   Port Key ....................... 5     Port Key ........................ 0
%   Port Priority .............. 32768     Port Priority ................... 0
%   Port Number ..,,,,,.......... 5001     Port Number ..................... 0
%   Mode ...................... Active     Mode ...................... Passive
%   Timeout ..................... Long     Timeout ..................... Short
%   Individual ................... Yes     Individual .................... Yes
%   Synchronised ................. Yes     Synchronised .................. Yes
%   Collecting ................... Yes     Collecting .................... Yes
%   Distributing ................. Yes     Distributing .................. Yes
%   Defaulted .................... Yes     Defaulted ..................... Yes
%   Expired ........................ No    Expired ........................ No
```

show static-channel-group

Use this command to display all configured static channel groups and their corresponding member ports. Note that a static channel group is the same as a static aggregator.

The command output also shows the thrash limiting status. If thrash limiting is detected and the **thrash limiting** parameter of the thrash-limiting command on page 15.50 is set to **vlan disable**, the output will also show the VLANs on which thrashing is detected.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax

```
show static-channel-group
```

Mode

User Exec and Privileged Exec

Example

```
awplus#   show static-channel-group
```

Output

Figure 21-9: Example output from the **show static-channel-group** command

```
% LAG Maximum        : 128
% LAG Static  Maximum: 96
% LAG Dynamic Maximum: 32
% LAG Static  Count  : 2
% LAG Dynamic Count  : 2
% LAG Total   Count  : 4
% Static Aggregator: sa2
% Member:
   port1.0.1
% Static Aggregator: sa3
% Member:
   port1.0.2
```

Related Commands

static-channel-group

static-channel-group

Use this command to create a static channel group, also known as a static aggregator, or add a member port to an existing static channel group.

You can create up to 96 static channel groups (and up to 32 dynamic channel groups).

Use the **no** variant of this command to remove the switch port from the static channel group.

Syntax

```
static-channel-group <static-channel-group-number>

no static-channel-group
```

Parameter	Description
<static-channel-group-number>	<1-96> Static channel group number.

Mode Interface Configuration

Usage This command adds the switch port to the static channel group with the specified channel group number. If the channel group does not exist, it is created, and the port is added to it. The **no** prefix detaches the port from the static channel group. If the port is the last member to be removed, the static channel group is deleted.

All the ports in a channel group must have the same VLAN configuration: they must belong to the same VLANs and have the same tagging status, and can only be operated on as a group.

Once the static channel group has been created, it is treated as a switch port, and can be referred to in other commands that apply to switch ports.

To refer to a static channel group in other static channel group commands, use the channel group number. To specify a static channel group in other commands, prefix the channel group number with **sa**. For example, 'sa3' refers to the static channel group with channel group number 3.

For more on static channel groups, see "Static Channel Groups" on page 20.4 and Chapter 20, Link Aggregation Introduction and Configuration.

Examples To define a static channel group on a switch port, use the commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.6
awplus(config-if)# static-channel-group 3
```

To reference the pre-defined static channel group 2 as an interface apply the example commands as below:

```
         awplus# configure terminal
  awplus(config)# interface port1.0.8
awplus(config-if)# static-channel-group 2
awplus(config-if)# exit
  awplus(config)# interface port.1.0.10
awplus(config-if)# static-channel-group 2
awplus(config-if)# exit
  awplus(config)# interface sa2
awplus(config-if)#
```

Related Commands show static-channel-group

undebug lacp

This command applies the functionality of the no debug lacp command on page 21.6.

Chapter 22: GVRP Introduction and Configuration

Introduction

GVRP enables the automatic VLAN configuration of switches in a network by allowing GVRP enabled switches to dynamically exchange VLAN configuration information with each other. GVRP is based on GARP, which defines how attributes, like VIDs, are registered and deregistered. This makes it easier to manage VLANs that span more than one switch. Without GVRP, you have to manually configure your switches to ensure that the various parts of the VLANs can communicate with each other across the different switches. With GVRP this is done for you automatically.

The switch uses GVRP protocol data units (PDUs) to share VLAN information among GVRP-active devices. The PDUs contain the VID numbers of all the VLANs on the switch. When the switch receives a GVRP PDU on a port, it examines the PDU to determine the VIDs of the VLANs on the device that sent it. It then does the following:

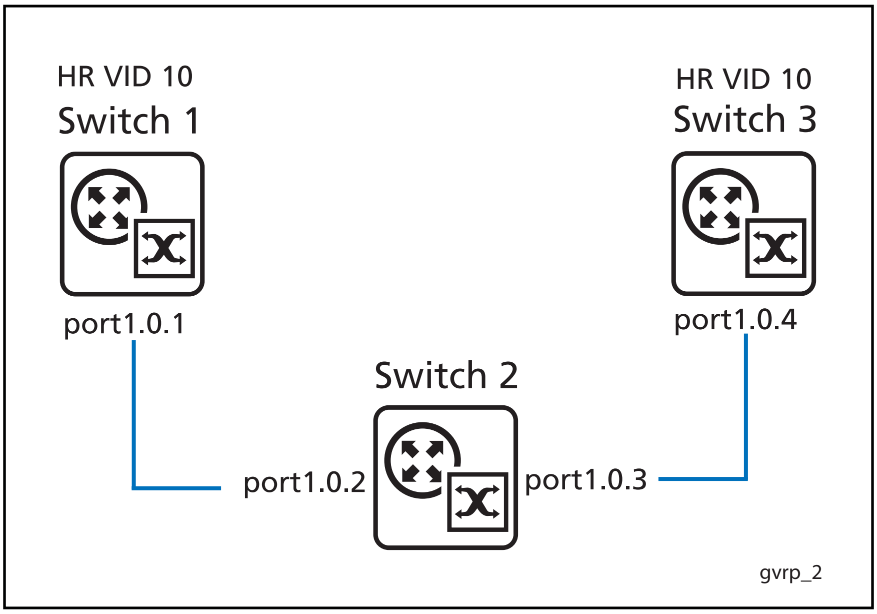
- If the PDU contains a VID of a VLAN that does not exist on the switch, it creates this VLAN and adds the port that received the PDU as a tagged member of the VLAN. A VLAN created by GVRP is called a dynamic GVRP VLAN.
- If the PDU contains a VID of a VLAN that already exists on the switch but the receiving port is not a member of it, the switch adds the port as a tagged member of the VLAN. A port that has been added by GVRP to a static VLAN (that is a user-created VLAN) is called a dynamic GVRP port.

Only GVRP can modify or delete dynamic GVRP VLANs. Dynamic GVRP VLANs exist only so long as the switch continues to receive GVRP PDUs that contain the VID of that VLAN. If there are no more relevant GVRP PDUs arriving, or there are no active links in the VLAN, GVRP deletes it from the switch.

A dynamic GVRP port in a static VLAN remains a member of the VLAN as long as the switch continues to receive GVRP PDUs that contain the VID of that VLAN. If the relevant GVRP PDUs are no longer being received on the port, then GVRP removes the dynamic port from the VLAN, but does not delete the VLAN if the VLAN is a static VLAN.

GVRP Example

The example consists of three switches. Switch 1 and Switch 3 have the HR VLAN 10, but Switch 2 is not configured with the HR VLAN 10. Consequently, the end nodes of the two parts of the HR VLAN 10 cannot communicate with each other because Switch 2 does not have VLAN 10.



Without GVRP, you would have to manually add the HR VLAN 10 to Switch 2. But with GVRP, the VLAN is added automatically. Here is how GVRP resolves this example.

1. Interface `port1.0.1` on Switch 1 sends a PDU (Protocol Data Unit) to interface `port1.0.2` on Switch 2 that contains the VIDs of all the VLANs on Switch 1, including VID 10 for the HR VLAN.

2. Switch 2 examines the PDU it receives on interface `port1.0.2` and finds that it does not have a VLAN with a VID 10. In response, it creates the VLAN as a dynamic GVRP VLAN, assigning it VID 10. Switch 2 then adds interface `port1.0.2`, the switch port that received the PDU, as a tagged member of HR VLAN 10.

3. Switch 2 sends a PDU from interface `port1.0.3` containing all the VIDs of the VLANs on the switch, including the new VID 10. Note at this point interface `port1.0.3` is not a member of VLAN 10. Ports are added to VLANs when they receive PDUs from other switches in the network, not when they transmit PDUs.

4. Switch 3 receives the PDU on interface port1.0.4 and, after examining it, finds that one of the VLANs on Switch 2 has the VID 10, which matches the VID of an already existing VLAN on the switch. So it does not create the VLAN because it already exists. It then determines whether the port that received the PDU, in this case interface `port1.0.4`, is a member of the VLAN. If it is not a member, it adds the port to the VLAN as a tagged dynamic GVRP port. If the port is already a member of the VLAN, then no change is made.

5. Switch 3 sends a PDU out interface `port1.0.4` to interface `port1.0.3` on Switch 2.

6. Switch 2 receives the PDU on interface `port1.0.3` and then adds the port as a tagged dynamic GVRP port to the dynamic GVRP VLAN 10.

There is now a communications path for the end nodes of the HR VLAN 10 on Switch 1 and Switch 3. GVRP created the new dynamic GVRP VLAN with a VID of 10 on Switch 2 and added interfaces `port1.0.2` and `port1.0.3` to HR VLAN 10 as tagged dynamic GVRP ports.

GVRP Guidelines

Here are the guidelines for configuring GVRP on your switch:

- All ports the constitute a network link between the switch and the other switches must be running GVRP.
- You cannot modify or delete dynamic GVRP VLANs.
- You cannot remove dynamic GVRP ports from static or dynamic VLANs.
- There is limit of 400 VLANs supported by the AlliedWare Plus GVRP implementation. VLANs may be numbered 1-4094, but a limit of 400 of these VLANs are supported.
- MSTP is not supported by the current AlliedWare Plus GVRP implementation. GVRP and MSTP are mutually exclusive. STP and RSTP are supported by GVRP.
- VCStack is not supported by the current AlliedWare Plus GVRP implementation.
- To be detected by GVRP, a VLAN must have at least one active port. GVRP cannot detect a VLAN that does not have any active nodes or valid port links.
- Rebooting the switch erases all dynamic GVRP VLANs and dynamic GVRP port assignments. The dynamic assignments are relearned by the switch as PDUs arrive on the ports from other switches.
- GVRP has three timers: join timer, leave timer, and leave all timer. The values for these timers must be set the same on all switches running GVRP. Timers with different values on different switches can result in GVRP compatibility problems.
- You can convert dynamic GVRP VLANs and dynamic GVRP port assignments to static VLANs and static port assignments.
- The default port settings on the switch for GVRP is inactive, meaning that the ports will not participate in GVRP until enabled on the switch globally and on the interface locally.
- Allied Telesis recommends disabling GVRP on those ports that are connected to GVRP-inactive devices, meaning any switches that do not have the GVRP feature enabled.
- PDUs are transmitted from only those switch ports where GVRP is enabled.
- Private VLAN trunk ports are not supported by the current AlliedWare Plus GVRP implementation. GVRP and private VLAN trunk ports are mutually exclusive.

GVRP and Network Security

GVRP should be used with caution because it can expose your network to unauthorized access. If a network intruder were to connect to a switch port running GVRP and transmit a bogus GVRP PDU containing VIDs of restricted VLANs, GVRP would make the port a member of the VLANs, giving the intruder access to restricted areas of your network.

Here are a few suggestions to protect against this type of unauthorized network intrusion:

- Activating GVRP only on those switch ports connected to other GVRP devices. Do not activate GVRP on ports that are connected to GVRP inactive devices.
- Converting all dynamic GVRP VLANs and dynamic GVRP ports to static assignments, and then turning off GVRP on all the switches. This preserves the new VLAN assignments while protecting against unauthorized network intrusion.

GVRP-inactive Intermediate Switches

If two GVRP-active devices are separated by a GVRP-inactive switch, the GVRP-active devices may not be able to share VLAN information. There are two issues involved.

The first is whether the intermediate switch forwards the GVRP PDUs that it receives from the GVRP-active switches. GVRP PDUs are management frames, intended for the switch's CPU. In all likelihood, a GVRP-inactive switch will discard the PDUs because it will not recognize them.

The second issue is that even if a GVRP-inactive switch forwards GVRP PDUs, it will not automatically create the VLANs. Consequently, even if GVRP-active switches receive the PDUs and create the necessary VLANs, an intermediate switch may block the VLAN traffic, unless you modify its VLANs and port assignments manually.

Enabling GVRP on the Switch

The command for enabling GVRP on the switch is found in the Global Configuration mode. It is the gvrp enable (global) command. After the command is entered, the switch immediately begins to transmit PDUs from those ports where GVRP is enabled.

Further, to enable the switch to create dynamic VLANs if it receives GVRP PDUs that contain VIDs for VLANs it does not currently have, use the command gvrp dynamic-vlan-creation.

Here are the commands to enable GVRP on the switch and enable to switch to create dynamic VLANs if it receives GVRP PDUs that contain VIDs for VLANs it does not currently have:

```
awplus>enable
awplus#configure terminal
awplus(config)#gvrp enable
awplus(config)#gvrp dynamic-vlan-creation
```

For reference information, refer to the gvrp enable (global) command and the gvrp dynamic-vlan-creation command in the GVRP Commands chapter.

Enabling GVRP on the Ports

To activate GVRP on the ports so that they transmit GVRP PDUs, use the gvrp registration and the gvrp (interface) commands in the Interface Configuration mode. Because the default setting for GVRP on the ports is disabled, you need to use these commands if you want to re-enable GVRP after disabling it on a port.

This example of these commands activates GVRP on interface `port1.0.12`, `port1.0.13`, and `port1.0.17`:

```
awplus>enable
awplus#configure terminal
awplus(config)#interface port1.0.12,port1.0.13,port1.0.17
awplus(config-if)#gvrp registration normal
awplus(config-if)#gvrp
```

For reference information, refer to the gvrp registration and gvrp (interface) commands in the GVRP Commands chapter.

Setting the GVRP Timers

The switch has a join timer, a leave timer, and a leave all timer. You should not change the timers unless you understand their functions. (Refer to the IEEE 802.1p standard for the timer definitions.) The timers have to be set the same on all GARP-active network devices and the join timer and the leave timer have to be set according to the following rule:

join timer <= (2 x (leave timer))

The commands for setting the timers are in the Interface Configuration mode. They are:

gvrp timer join
gvrp timer leave
gvrp timer leaveall

The timers are set in one hundredths of a second. This example sets the join timer to 0.2 seconds, the leave timer to 0.8 seconds and the leave all timer to 10 seconds for port1.0.2:

```
awplus>enable
awplus#configure terminal
awplus(config)#interface port1.0.2
awplus(config-if)#gvrp timer join 20
awplus(config-if)#gvrp timer leave 80
awplus(config-if)#gvrp timer leaveall 1000
```

For reference information, refer to gvrp timer command in the GVRP Commands chapter.

Disabling GVRP on the Ports

To disable GVRP on the ports, use the gvrp registration none and no gvrp (interface) commands in the Interface Configuration mode.

This example of the command deactivates GVRP on interfaces `port1.0.4` and `port1.0.5`:

```
awplus>enable
awplus#configure terminal
awplus(config)#interface port1.0.4,port1.0.5
awplus(config-if)#gvrp registration none
awplus(config-if)#no gvrp
```

For reference information, refer to gvrp registration and gvrp (interface) command in the GVRP Commands chapter.

Disabling GVRP on the Switch

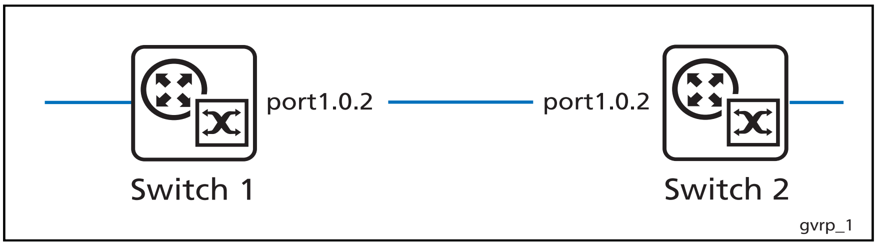
To disable GVRP to stop the switch from learning any further dynamic VLANs or GVRP ports, use the no gvrp (interface) enable command in the Global Configuration mode. Here is the command.

```
awplus>enable
awplus#configure terminal
awplus(config)#no gvrp enable
```

For reference information, refer to the gvrp (interface) command in the GVRP Commands chapter.

Configuring and validating GVRP

GVRP (GARP VLAN Registration Protocol) allows the exchange of VLAN information between switches in a network. If one switch is manually configured with multiple VLANs, other switches in the network learn about these VLANs dynamically through GVRP.



Switch 1: Configuring GVRP to receive VLANs from Switch 1

awplus# `configure terminal`	Enter the Global Configuration mode from the Privileged Exec mode.
awplus(config)# `gvrp enable`	Enter GVRP on **Switch 1**.
awplus(config)# `gvrp dynamic-vlan-creation`	Enable dynamic VLAN creation for GVRP. Note that GVRP is now enabled globally for **Switch 1**.
awplus(config)# `interface port1.0.2`	Specify an interface (`port1.0.2`) to be configured and enter Interface Configuration mode.
awplus(config-if)# `switchport mode trunk`	Set the switching characteristics of the interface as trunk and specify tagged frames only. Any frames not tagged as trunk frames are discarded.
awplus(config-if)# `switchport trunk allowed vlan all`	Apply to all VLANs on this interface.
awplus(config-if)# `gvrp`	Enable GVRP on switch port `port1.0.2`. Note that GVRP is now set up on interface `port1.0.2` as GVRP is also enabled globally for **Switch 1**.
awplus(config-if)# `exit`	Exit Interface Configuration mode and enter Global Configuration mode.
awplus(config)# `exit`	Exit Global Configuration mode and enter Privileged Exec mode.
awplus# `show gvrp configuration`	Show GVRP configuration on **Switch 1** to confirm GVRP is ready to propagate VLANs.

Switch 2: Configuring GVRP & creating VLANs to propagate:

awplus# enable	Enter the Privileged Exec mode.
awplus# configure terminal	Enter the Global Configuration mode.
awplus(config)# gvrp enable	Enter GVRP on **Switch 2**.
awplus(config)# vlan database	Create VLANs to propagate between **Switch 1** and **Switch 2** with GVRP enabled on the Switches and on the interfaces on each Switch.
awplus(config-vlan)# vlan 20-30	Create 11 VLANs with VIDs 20 through 30 to propagate between interface `port1.0.2` on **Switch 1** and **Switch 2**.
awplus(config)# gvrp dynamic-vlan-creation	Enable dynamic VLAN creation for GVRP. Note that GVRP is now enabled globally for **Switch 2**.
awplus(config)# interface port1.0.2	Specify an interface (`port1.0.2`) to be configured and enter Interface Configuration mode.
awplus(config-if)# switchport mode trunk	Set the switching characteristics of the interface as trunk and specify tagged frames only. Any frames not tagged as trunk frames are discarded.
awplus(config-if)# switchport trunk allowed vlan all	Set this interface to be a tagged member of all VLANs.
awplus(config-if)# gvrp	Enable GVRP on switch port `port1.0.2`.
awplus(config-if)# exit	Exit Interface Configuration mode and enter Global Configuration mode.
awplus(config)# exit	Exit Global Configuration mode and enter Privileged Exec mode.
awplus# show gvrp configuration	Show GVRP configuration on **Switch 2** to confirm GVRP is ready to propagate VLANs.

Switch 1: Validating VLANs have propagated from Switch 2:

`awplus#`	
`show vlan`	Confirm the VLANs are available from **Switch 2** on **Switch 1** by examining show output to confirm VLANs from **Switch 2** are on **Switch 1**.

Names of Commands Used

gvrp (interface)
gvrp dynamic-vlan-creation
switchport mode trunk
vlan database
vlan

Validation Commands

show vlan

Chapter 23: GVRP Commands

Command List

With GVRP enabled the switch can exchange VLAN configuration information with other GVRP enabled switches. VLANs can be dynamically created and managed through trunk ports.

- There is limit of 400 VLANs supported by the AlliedWare Plus GVRP implementation. VLANs may be numbered 1-4094, but a limit of 400 of these VLANs are supported.
- MSTP is not supported by the AlliedWare Plus GVRP implementation. GVRP and MSTP are mutually exclusive. STP and RSTP are supported by GVRP.
- VCStack is not supported by the current AlliedWare Plus GVRP implementation.

This chapter provides an alphabetical reference for commands used to configure GVRP. For information about GVRP, including configuration, see Chapter 22, GVRP Introduction and Configuration.

clear gvrp statistics

Use this command to clear the GVRP statistics for all switchports, or for a specific switchport.

Syntax

```
clear gvrp statistics {all|<interface>}
```

Parameter	Description
`all`	Specify all switchports to clear GVRP statistics.
`<interface>`	Specify the switchport to clear GVRP statistics.

Mode Privileged Exec

Usage Use this command together with the show gvrp statistics command to troubleshoot GVRP.

Examples To clear all GVRP statistics for all switchport on the switch, enter the following command:

```
awplus#clear gvrp statistics all
```

To clear GVRP statistics for switchport interface `port1.0.3`, enter the following command:

```
awplus#clear gvrp statistics port1.0.3
```

Related Commands show gvrp statistics

debug gvrp

Use this command to debug GVRP packets and commands, sending output to the console.

Use the **no** variant of this command to turn off debugging for GVRP packets and commands.

Syntax

```
debug gvrp {all|cli|event|packet}

no debug gvrp {all|cli|event|packet}
```

Parameter	Description
all	Specifies debugging for all levels.
cli	Specifies debugging for commands.
event	Specified debugging for events.
packet	Specifies debugging for packets.

Mode Privileged Exec and Global Configuration

Examples To send debug output to the console for GVRP packets and GVRP commands, and to enable the display of debug output on the console first, enter the following commands:

```
        awplus#terminal monitor
        awplus#configure terminal
awplus(config)#debug gvrp all
```

To send debug output for GVRP packets to the console, enter the following commands:

```
        awplus#terminal monitor
        awplus#configure terminal
awplus(config)#debug gvrp packets
```

To send debug output for GVRP commands to the console, enter the following commands:

```
        awplus#terminal monitor
        awplus#configure terminal
awplus(config)#debug gvrp cli
```

To stop sending debug output for GVRP packets and GVRP commands to the console, and to stop the display of any debug output on the console, enter the following commands:

```
        awplus#terminal no monitor
        awplus#configure terminal
awplus(config)#no debug gvrp all
```

Related Commands show debugging gvrp
terminal monitor

gvrp (interface)

Use this command to enable GVRP for switchport interfaces.

Use the **no** variant of this command to disable GVRP for switchport interfaces.

Syntax

```
gvrp

no gvrp
```

Mode Interface Configuration (for switchport interfaces).

Default Disabled by default.

Usage Use this command to enable GVRP on switchport interfaces. Note this command does not enable GVRP for the switch. To enable GVRP on switchports use this command in Interface Configuration mode. You must issue a gvrp enable (global) command before issuing a gvrp (interface) command.

You must enable GVRP on both ends of a link for GVRP to propagate VLANs between links.

Note MSTP is not supported by the current AlliedWare Plus GVRP implementation. GVRP and MSTP are mutually exclusive. STP and RSTP are supported by GVRP.

Private VLAN trunk ports are not supported by the current AlliedWare Plus GVRP implementation. GVRP and private VLAN trunk ports are mutually exclusive.

Examples To enable GVRP on interfaces `port1.0.1-port1.0.2`, enter the following commands:

```
          awplus#configure terminal
  awplus(config)#gvrp enable
  awplus(config)#interface port1.0.1-port1.0.2
awplus(config-if)#gvrp
```

To disable GVRP on interfaces `port1.0.1-port1.0.2`, enter the following commands:

```
          awplus#configure terminal
  awplus(config)#interface port1.0.1-port1.0.2
awplus(config-if)#no gvrp
```

Validation Commands show gvrp configuration

Related Commands gvrp dynamic-vlan-creation
gvrp enable (global)

gvrp dynamic-vlan-creation

Use this command to enable dynamic VLAN creation globally for the switch.

Use the **no** variant of this command to disable dynamic VLAN creation globally for the switch.

Syntax

```
gvrp dynamic-vlan-creation
no gvrp dynamic-vlan-creation
```

Mode Global Configuration

Default Disabled by default.

Usage You must enable GVRP on both ends of a link for GVRP to propagate VLANs between links.

You must also enable GVRP globally in Global Configuration mode before enabling GVRP on an interface in Interface Configuration mode. Both of these tasks must occur to create VLANs.

Note There is limit of 400 VLANs supported by the AlliedWare Plus GVRP implementation. VLANs may be numbered 1-4094, but a limit of 400 of these VLANs are supported.

.Examples To enable GVRP dynamic VLAN creation on the switch, enter the following commands:

```
        awplus#configure terminal
awplus(config)#gvrp enable
awplus(config)#gvrp dynamic-vlan-creation
```

Enter the following commands for switches with hostnames `awplus_switch1` and `awplus_switch2` respectively, so `awplus_switch1` propagates VLANs to `awplus_switch2` and `awplus_switch2` propagates VLANs to `awplus_switch1`:

```
        awplus_switch1#configure terminal
awplus_switch1(config)#gvrp enable
awplus_switch1(config)#gvrp dynamic-vlan-creation

        awplus_switch2#configure terminal
awplus_switch2(config)#gvrp enable
awplus_switch2(config)#gvrp dynamic-vlan-creation
```

To disable GVRP dynamic VLAN creation on the switch, enter the following commands:

```
        awplus#configure terminal
awplus(config)#no gvrp dynamic-vlan-creation
```

Validation Commands show gvrp configuration

Related Commands gvrp enable (global)

gvrp enable (global)

Use this command to enable GVRP globally for the switch.

Use the **no** variant of this command to disable GVRP globally for the switch.

Syntax

```
gvrp enable
no gvrp enable
```

Mode Global Configuration

Default Disabled by default.

Usage Use this command to enable GVRP on the switch. Note that this command does not enable GVRP on switchports. To enable GVRP on switchports use the gvrp (interface) command in Interface Configuration mode. You must issue a gvrp enable (global) command before issuing a gvrp (interface) command.

You must enable GVRP on both ends of a link for GVRP to propagate VLANs between links.

Note MSTP is not supported by the current AlliedWare Plus GVRP implementation. GVRP and MSTP are mutually exclusive. STP and RSTP are supported by GVRP.

Private VLAN trunk ports are not supported by the current AlliedWare Plus GVRP implementation. GVRP and private VLAN trunk ports are mutually exclusive.

Examples To enable GVRP for the switch, before enabling GVRP on switchports, enter the following commands:

```
        awplus#configure terminal
awplus(config)#gvrp enable
```

To disable GVRP on the switch, which will also disable GVRP enabled on switchports, enter the following commands:

```
        awplus#configure terminal
awplus(config)#no gvrp enable
```

Validation Commands show gvrp configuration

Related Commands gvrp (interface)
gvrp dynamic-vlan-creation

gvrp registration

Use this command to set GVRP registration to normal, fixed, and forbidden registration modes.

Syntax `gvrp registration {normal|fixed|forbidden}`

Parameter	Description
`normal`	Specify dynamic GVRP registration and deregistration of VLANs.
`fixed`	Specify fixed GVRP registration and deregistration of VLANs.
`forbidden`	Specify no GVRP registration of VLANs. VLANs are deregistered.

Mode Interface Configuration

Default Normal registration is the default.

Usage Configuring a trunk port in normal registration mode allows dynamic creation of VLANs. Normal mode is the default mode. Validate using the show gvrp configuration command.

Configuring a trunk port in fixed registration mode allows manual creation of VLANs.

Configuring a trunk port in forbidden registration mode prevents VLAN creation on the port.

Examples To configure GVRP registration to `fixed` on `port1.0.1`, enter these commands:

```
          awplus#configure terminal
  awplus(config)#interface port1.0.1
awplus(config-if)#gvrp registration fixed
```

To configure GVRP registration to `normal` on `port1.0.2`, enter these commands:

```
          awplus#configure terminal
  awplus(config)#interface port1.0.2
awplus(config-if)#gvrp registration normal
```

To configure GVRP registration to `forbidden` on `port1.0.3`, enter these commands:

```
          awplus#configure terminal
  awplus(config)#interface port1.0.3
awplus(config-if)#gvrp registration forbidden
```

Validation Commands show gvrp configuration

gvrp timer

Use this command to set GVRP timers in Interface Configuration mode for a given interface.

Use the **no** variant of this command to reset the GVRP timers to the defaults specified in the table below.

Syntax

```
gvrp timer
   {join <timer-value>|leave <timer-value>|leaveall <timer-value>}

no gvrp timer {join|leave|leaveall}
```

Parameter	Description
join	Specifies the timer for joining the group (default is 20 centiseconds / hundredths of a second, or 200 milliseconds).
leave	Specifies the timer for leaving a group (default is 60 centiseconds / hundredths of a second, or 600 milliseconds).
leaveall	Specifies the timer for leaving all groups (default is 1000 centiseconds / hundredths of a second, or 10,000 milliseconds).
<*timer-value*>	<1-65535> The timer value in hundredths of a second (centiseconds).

Mode Interface Configuration

Defaults The default join time value is 20 centiseconds (200 milliseconds), the default leave timer value is 60 centiseconds (600 milliseconds), and the default leaveall timer value is 1000 centiseconds (10,000 milliseconds).

Usage When configuring the `join` timer, set it to less than or equal to twice the `leave` timer value. The settings for the `join` and `leave` timers must be the same for all GVRP enabled switches.

Use the show gvrp timer command to confirm GVRP timers set with this command.

Examples To set the GVRP `join` timer to 300 hundredths of a second for interface `port1.0.1`, enter the following commands:

```
        awplus#configure terminal
awplus(config)#interface port1.0.1
awplus(config-if)#gvrp timer join 20
```

To set the GVRP `leave` timer to 600 hundredths of a second for interface `port1.0.2`, enter the following commands:

```
        awplus#configure terminal
awplus(config)#interface port1.0.2
awplus(config-if)#gvrp timer leave 60
```

To set the GVRP `leaveall` timer to 1000 hundredths of a second for interface `port1.0.1`, enter the following commands:

```
           awplus#configure terminal
   awplus(config)#interface port1.0.1
awplus(config-if)#gvrp timer leaveall 1000
```

To reset the GVRP `join` timer to its default (200 milliseconds) for interface `port1.0.1`, enter the following commands:

```
           awplus#configure terminal
   awplus(config)#interface port1.0.1
awplus(config-if)#no gvrp timer join
```

To reset the GVRP `leave` timer to its default (600 milliseconds) for interface `port1.0.2`, enter the following commands:

```
           awplus#configure terminal
   awplus(config)#interface port1.0.2
awplus(config-if)#no gvrp timer leave
```

To reset the GVRP `leaveall` timer to its default (10,000 milliseconds) for interface `port1.0.3`, enter the following commands:

```
           awplus#configure terminal
   awplus(config)#interface port1.0.3
awplus(config-if)#no gvrp timer leaveall
```

Related Commands show gvrp timer

show debugging gvrp

Use this command to display the GVRP debugging option set.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax `show debugging gvrp`

Mode User Exec and Privileged Exec

Example Enter the following commands to display GVRP debugging output on the console:

```
awplus#configure terminal
awplus(config)#debug gvrp all
awplus(config)#exit
```

Output See sample output from the `show debugging gvrp` after entering `debug gvrp all`:

```
GVRP debugging status:
  GVRP Event debugging is on
  GVRP CLI debugging is on
  GVRP Timer debugging is on
  GVRP Packet debugging is on
```

Related Commands debug gvrp

show gvrp configuration

Use this command to display GVRP configuration data for a switch.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax

```
show gvrp configuration
```

Mode

User Exec and Privileged Exec

Example

To show GVRP configuration for the switch, enter the following command:

```
awplus#show gvrp configuration
```

Output

The following is an output of this command displaying the GVRP configuration for a switch:

```
awplus#show gvrp configuration
Global GVRP Configuration:
GVRP Feature: Enabled
Dynamic Vlan Creation: Disabled
Port based GVRP Configuration:
                                            Timers(centiseconds)
Port    GVRP Status Registration  Applicant   Join   Leave
LeaveAll
----------------------------------------------------------------
port1.0.1 Enabled   Normal        Normal      20       60    1000
port1.0.2 Enabled   Normal        Normal      200     600   10000
```

show gvrp machine

Use this command to display the state machine for GVRP.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax `show gvrp machine`

Mode User Exec and Privileged Exec

Example To show the GVRP state machine for the switch, enter the following command:

```
awplus#show gvrp machine
```

Output See the following output of this command displaying the GVRP state machine.

```
awplus show gvrp machine
  port = 1.0.1  applicant state = QA   registrar state = INN
  port = 1.0.2  applicant state = QA   registrar state = INN
```

show gvrp statistics

Use this command to display a statistical summary of GVRP information for the switch.

For information on output options, see “Controlling “show” Command Output” on page 1.41.

Syntax `show gvrp statistics [<interface>]`

Parameter	Description
<interface>	The name of the switchport interface.

Mode User Exec and Privileged Exec

Usage Use this command together with the clear gvrp statistics command to troubleshoot GVRP.

Examples To show the GVRP statistics for all switchport interfaces, enter the following command:

```
awplus#show gvrp statistics
```

To show the GVRP statistics for switchport interfaces `port1.0.1` and `port1.0.2`, enter the following command:

```
awplus#show gvrp statistics port1.0.1-port1.0.2
```

Output The following is an output of this command displaying a statistical summary for `port1.0.1-port1.0.2`

```
awplus# show gvrp statistics port1.0.1-port1.0.2
Port      JoinEmpty    JoinIn   LeaveEmpty     LeaveIn     Empty
-----------------------------------------------------------------
1.0.1     RX           0          2           0          0        0
          TX           0          0           0          0        0
1.0.2     RX           0          1           0          0        1
          TX           0          0           0          0        0
```

Related Commands clear gvrp statistics

show gvrp timer

Use this command to display data for the GVRP timers set with the gvrp timer command.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax

```
show gvrp timer <interface>
```

Parameter	Description
<interface>	The name of the switchport interface.

Mode

User Exec and Privileged Exec

Examples

To show the GVRP timers for all switchport interfaces, enter the following command:

```
awplus#show gvrp timer
```

To show the GVRP timers for switchport interface `port1.0.1`, enter the following command:

```
awplus#show gvrp timer port1.0.1
```

Output

The following show output displays data for timers on the switchport interface `port1.0.1`

```
awplus# show gvrp timer port1.0.1
Timer           Timer Value (centiseconds)
------------------------------------------
Join                 20
Leave                60
Leave All            1000
```

Related Commands

gvrp timer

Part 3: Layer Three, Switching and Routing

Chapter 24: Internet Protocol (IP) Addressing and Protocols

Introduction

This chapter describes how to configure IPv4 addressing and the protocols used to help IP function on your network.

As well as the familiar Internet, with uppercase “I”, the term internet (with lowercase “i”) can refer to any network (usually a wide area network) that uses the Internet Protocol. This chapter concentrates on this definition—a generalized network that uses IP as its transport protocol.

Assigning an IP Address

To configure your device to perform IP routing (for example, to access the Internet) you need to configure IP. You also need to configure IP if you want to manage your device from any IP-based management process (such as SSH, Telnet, or SNMP).

Add an IP address to each of the interfaces that you want to process IP traffic.

You can configure an interface on your device with a static IP address, or with a dynamic IP address assigned using your device’s DHCP client.

Static IP addresses

To add a static IP address to an interface, enter interface mode for the interface that you want to configure, then use the command:

```
awplus(config-if)# ip address <ip-addr/prefix-length>
                    [secondary [label <label>]]
```

where <ip-address/m> the IP address followed by a slash then the prefix length. Note that you cannot specify the mask in dotted decimal notation in this command.

For example, to give the interface vlan1 an address of 192.168.10.10, with a class C subnet mask, use the command:

```
awplus(config-if)# ip address 192.168.10.10/24
```

The **secondary** parameter allows you to add multiple IP addresses to an interface using this command. Each interface must have a primary IP address before you can add a secondary address. Your device treats secondary addresses the same as primary addresses in most instances, such as responding for ARP requests for the IP address. However, the only packets generated that have a secondary address as source address are routing updates. You can define up to 32 secondary addresses on a single interface.

DHCP dynamic addresses

When you use the DHCP client, it obtains the IP address and subnet mask for the interface, and other IP configuration parameters, from a DHCP server. To configure an interface to gain its IP configuration using the DHCP client, use the command:

```
awplus(config-if)# ip address dhcp [client-id <interface>]
                    [hostname <hostname>]
```

If an IP interface is configured to get its IP address and subnet mask from DHCP, the interface does not take part in IP routing until the IP address and subnet mask have been set by DHCP.

If you need to make a static entry in your DHCP server for the device, you need your device's MAC address, which you can display by using the command:

```
awplus# show interface
```

See Chapter 60, Dynamic Host Configuration Protocol (DHCP) Introduction for more information about DHCP.

Address Resolution Protocol (ARP)

Address Resolution Protocol (ARP) is used by your device to dynamically learn the Layer 2 address of devices in its networks. Most hosts also have a MAC physical address in addition to the assigned IP address. For Ethernet, this is a 6-byte, globally unique number. ARP enables your device to learn the physical address of the host that has a given IP address.

When your device needs to forward packets to a destination that it does not know the Layer 2 address of, it broadcasts an ARP request to determine where to send the packet. The ARP request is a broadcast packet and includes the target IP address. All stations on the LAN receive this broadcast but only one host recognizes its own IP address. It replies, thereby giving your device its physical address.

Your device creates a dynamic ARP entry in its ARP cache, to record the IP address to physical address mapping (also called a binding). It uses that ARP entry to forward further packets to that address.

The ARP protocol is described in RFC 826, ***An Ethernet Address Resolution Protocol—or—Converting Network Protocol Addresses to 48 bit Ethernet Address for Transmission on Ethernet Hardware***.

Static ARP Entries

If your LAN includes hosts that do not support ARP, you can add a static ARP entry to the cache. However, it is rarely necessary to add an ARP entry this way. To add a static ARP entry, use the command:

Timing Out ARP Entries

Your device times out dynamic ARP entries to ensure that the cache does not fill with entries for hosts that are no longer active. If your device stops receiving traffic for a device specified in a dynamic ARP entry, it deletes the ARP entry after a configurable timeout period. Static ARP entries are not aged or automatically deleted.

Increasing the ARP timeout reduces the amount of network traffic. Decreasing the timeout makes your device more responsive to changes in network topology.

To set a timeout period, enter the interface mode, then use the command:

```
awplus(config-if)# arp-aging-timeout <0-432000>
```

Deleting ARP Entries

To remove a static ARP entry, use the command:

To clear the ARP cache of dynamic entries, use the command:

```
awplus# clear arp-cache
```

This removes the dynamic ARP entries for all interfaces.

To display the entries in the ARP cache, use the command:

```
awplus)# show arp
```

The ARP cache will be repopulated by the normal ARP learning mechanism. As long as the entries are relearned quickly enough, deleting dynamic ARP entries does not affect:

- the TCP/UDP connection status
- VRRP status

Proxy ARP

Proxy ARP (defined in RFC 1027) allows hosts that do not support routing (i.e. they have no knowledge of the network structure) to determine the physical addresses of hosts on other networks. Your device intercepts ARP broadcast packets and substitutes its own physical address for that of the remote host. This occurs only if your device has the best route to the remote host. By responding to the ARP request, your device ensures that subsequent packets from the local host are directed to its physical address, and it can then forward these to the remote host. The process is symmetrical.

Proxy ARP is disabled by default. To enable proxy ARP on an interface, use the commands:

```
   awplus(config)# interface <interface>
awplus(config-if)# ip proxy-arp
```

To disable Proxy ARP on an interface, use the command:

```
awplus(config-if)# no ip proxy-arp
```

To check Proxy ARP is enabled on an interface, use the **show running-config** command. If Proxy ARP has been enabled an entry shows **ip proxy-arp** below the interface it is enabled on. No **ip proxy-arp** entry below an interface in the config indicates Proxy ARP is disabled on it.

See the sample configuration commands and validation command with resulting output showing proxy ARP **enabled** on VLAN 2 below:

```
awplus#configure terminal
awplus(config)#interface vlan2
awplus(config-if)#ip proxy-arp
awplus(config-if)#end
awplus(config)#exit
awplus#show running-config
!
interface vlan2
 ip proxy-arp
 ip address 192.168.2.2/24
!
```

See the sample configuration commands and validation command with resulting output showing proxy ARP **disabled** on VLAN 2 below:

```
awplus#configure terminal
awplus(config)#interface vlan2
awplus(config-if)#no ip proxy-arp
awplus(config-if)#end
awplus(config)#exit
awplus#show running-config
!
interface vlan2
 ip address 192.168.2.2/24
!
```

Local Proxy ARP

Local Proxy ARP lets you stop MAC address resolution between hosts within an interface's subnet. This ensures that devices within a subnet cannot send traffic that bypasses Layer 3 routing on your device. This lets you monitor, filter, and control traffic between devices in the same subnet.

Local Proxy ARP extends proxy ARP by intercepting and responding to ARP requests between hosts within a subnet. Local proxy ARP responds to ARP requests with your device's own MAC address details instead of those from the destination host. This stops hosts from learning the MAC address of other hosts within its subnet.

When Local Proxy ARP is operating on an interface, your device does not generate or forward any ICMP-Redirect messages on that interface.

Local Proxy ARP is disabled by default. To enable local proxy ARP on an interface, use the commands:

```
   awplus(config)# interface <interface>
awplus(config-if)# ip local-proxy-arp
```

To disable local proxy ARP on an interface, use the command:

```
awplus(config-if)# no ip local-proxy-arp
```

To check Local Proxy ARP is enabled on an interface, use the **show running-config** command. If Local Proxy ARP has been enabled an entry shows **ip local-proxy-arp** below the interface it is enabled on. No **ip local-proxy-arp** entry below an interface in the config indicates Local Proxy ARP is disabled on it.

See the sample configuration commands and validation command with resulting output showing local proxy ARP **enabled** on VLAN 1 below:

```
awplus#configure terminal
awplus(config)#interface vlan1
awplus(config-if)#ip local-proxy-arp
awplus(config-if)#end
awplus(config)#exit
awplus#show running-config
!
interface vlan1
 ip local-proxy-arp
 ip address 192.168.1.2/24
!
```

See the sample configuration commands and validation command with resulting output showing Local Proxy ARP **disabled** on VLAN 1 below:

```
awplus#configure terminal
awplus(config)#interface vlan1
awplus(config-if)#no ip local-proxy-arp
awplus(config-if)#end
awplus(config)#exit
awplus#show running-config
!
interface vlan1
 ip address 192.168.1.2/24
!
```

ARP Logging

You can enable your device to log static and dynamic ARP entries, and you can select either default hexadecimal notation (HHHH.HHHH.HHHH) or standard IEEE format hexadecimal notation (HH-HH-HH-HH-HH-HH) for the MAC addresses displayed in the ARP log output.

If this feature is enabled, ARP log messages are stored on the device in RAM. If the device is rebooted the ARP log messages are lost. ARP logging is disabled by default.

To enable ARP logging, use the command:

```
awplus(config)# arp log [mac-address-format ieee]
```

You can specify whether the MAC address is displayed in the default hexadecimal notation HHHH.HHHH.HHHH or in the standard IEEE format HH-HH-HH-HH-HH-HH.

To disable ARP logging, use the command:

```
awplus(config)# no arp log [mac-address-format ieee]
```

To display the ARP log messages, use the command:

```
awplus(config)# show log | include ARP_LOG
```

See the sample ARP log output and descriptions of the fields displayed in the sample ARP log output in the arp log command on page 25.5.

Domain Name System (DNS)

The Domain Name System allows you to access remote systems by entering human-readable device host names rather than IP addresses. DNS works by creating a mapping between a device name, such as "www.alliedtelesis.com", and its IP address. These mappings are held on DNS servers. DNS translates meaningful domain names into IP addresses for networking equipment to locate and address these devices. The benefits of DNS are that domain names:

- can map to a new IP address if the host's IP address changes
- are easier to remember than an IP address
- allow organizations to use a domain name hierarchy that is independent of any IP address assignment

Your AlliedWare Plus^{TM} device has a the ability to resolve domain names for internally generated commands (DNS Client) as well as providing the DNS information to connected hosts (via DNS Relay, DHCP Server or DHCP Relay).

The DNS Client is enabled automatically when at least one DNS server is present on the interface. This client allows you to use domain names instead of IP addresses when using commands on your device from this interface.

The DNS Relay provides the presence of a local virtual DNS server which can service DNS lookup requests sent to it from local hosts. The DHCP Server can be configured to provide domain names information to DHCP clients during the lease process.

Domain name parts

Domain names are made up of a hierarchy of two or more name segments. Each segment is separated by a period. The format of domain names is the same as the host portion of a URL (Uniform Resource Locator). The first segment from the left is unique to the host, with each following segment mapping the host in the domain name hierarchy. The segment on the far right is a top-level domain name shared by many hosts.

Server hierarchy

A network of domain name servers maintains the mappings between domain names and their IP addresses. This network operates in a hierarchy that is similar to the structure of the domain names. When a local DNS server cannot resolve your request it sends the request to a higher level DNS server.

For example, to access the site "alliedtelesis.com", your PC sends a DNS enquiry to its local DNS server asking for the IP address matching alliedtelesis.com. If this address is already locally cached (following its recent use), the DNS server returns the IP address that matches alliedtelesis.com. If the DNS server does not have this address cached, it forwards the request upwards through the hierarchy of DNS servers until a DNS server can resolve the mapping. This means an often-used domain name is resolved quickly, while an uncommon or nonexistent domain may take longer to resolve or fail.

As well as the hierarchy of domain name servers accessible through the Internet, you can operate your own DNS server to map to private IP addresses within your network.

The DHCP server IP address can be either statically defined, or can be dynamically assigned via DHCPv4 option 6 using "ip name-server" on page 25.39 and DHCP option 15 using "ip domain-name" on page 25.22 if DHCP client is configured. See Chapter 60, Dynamic Host Configuration Protocol (DHCP) Introduction for more information about DHCP and DHCP options.

DNS Client

Your AlliedWare Plus^{TM} device has a DNS Client that is enabled automatically when you add a DNS server to your device. This client allows you to use domain names instead of IP addresses when using commands on your device.

To add a DNS server to the list of servers that the device sends DNS queries to, use the command:

```
awplus(config)# ip name-server <ip-addr>
```

To check the list of servers that the device sends DNS queries to, use the command:

```
awplus# show ip name-server
```

To add a default domain name used to append to DNS requests, use the command:

```
awplus(config)# ip domain-name <domain-name>
```

For example, to use DNS to match hostnames to your internal network "example.net", use the command:

```
awplus(config)# ip domain-name example.net
```

If you then use the command **ping host2**, your device sends a DNS request for host2.example.net. To check the domain name configured with this command, use the command:

```
awplus# show ip domain-name
```

Alternatively you can create a list of domain names that your device will try in turn by using the command:

```
awplus(config)# ip domain-list <domain-name>
```

For example, to use DNS to match incomplete hostnames to the top level domains ".com", and ".net", use the commands:

```
awplus(config)# ip domain-list .com
awplus(config)# ip domain-list .net
```

If you then use the command **ping alliedtelesis**, your device sends a DNS request for alliedtelesis.com and if no match was found your device would then try alliedtelesis.net. To check the entries in the domain list, use the command:

```
awplus# show ip domain-list
```

To disable the DNS client on your device, use the command:

```
awplus(config)# no ip domain-lookup
```

To check the status of the DNS Client on your device, and the configured servers and domain names, use the command:

```
awplus# show hosts
```

DNS Relay

DNS Relay provides the presence of a local virtual DNS server on your AlliedWare Plus^{TM} device which can service DNS lookup requests sent to it from local hosts. The DNS Relay will usually relay the requests to an external, or upstream, DNS server. By default, DNS Relay is disabled.

Optionally, DNS name resolver caching may be enabled on the DNS Relay, which can provide some lookup speed advantage and avoid unnecessary repeated requests to external DNS servers. By default, DNS caching is disabled.

When the DNS Relay name resolver cache is enabled on your switch, the switch will maintain a cache of recently used mappings between domain names and IP addresses so that other identical requests can be responded to without further reference to an external, or upstream DNS server. When the switch receives a DNS query from a client the switch will attempt to match the request with entries in this cache. If the switch does not have this address cached, it forwards the request upwards through the hierarchy of DNS servers for resolution. The DNS cache has a limited size, and times out entries after a specified period of up to 60 minutes.

The relaying of DNS queries is required for use in networks where the DNS server and the clients connected to the switch are on different subnets and do not know how to reach each other.

DNS Relay uses the DNS server list configured by the **ip name-server** command to forward DNS query packets. To enable DNS Relay you need to configure the list of servers that the device sends DNS queries to and then enable DNS forwarding, as shown in the following example for a DNS server with an IPv4 address:

```
        awplus# configure terminal
awplus(config)# ip name-server 192.168.1.1
awplus(config)# ip name-server 192.168.1.2
awplus(config)# ip dns forwarding
```

Note both IPv4 and IPv6 support DNS record types. IPv4 and IPv6 are supported in DNS name-to-address and DNS address-to-name lookup processes. Specifying a name server and enabling DNS forwarding maps both IPv4 and IPv6 addresses. You can configure DNS Relay to use IPv6 addresses using the same commands used to configure DNS Relay to use IPv4 addresses, as shown in the following example:

```
        awplus# configure terminal
awplus(config)# ip name-server 2001:0db8:010d::1
awplus(config)# ip name-server 2001:0db8:010d::2
awplus(config)# ip dns forwarding
```

You can then configure DNS Relay behavior with the following commands:

To set the number of times the switch will retry to forward DNS queries, use the command:

```
awplus(config)# ip dns forwarding retry <0-100>
```

To set the number of seconds to wait for a response, use the command:

```
awplus(config)# ip dns forwarding timeout <0-3600>
```

To set the interface to use for forwarding and receiving DNS queries, use the command:

```
awplus(config)# ip dns forwarding source-interface
                <interface-name>
```

To specify the DNS Relay name resolver cache size and lifetime, use the command:

```
awplus(config)# ip dns forwarding cache [size <0-1000>]
                [timeout <60-3600>]
```

To remove entries from the DNS Relay name resolver cache, use the command:

```
awplus(config)# clear ip dns forwarding cache
```

Information which may be useful for troubleshooting DNS Relay is available using the DNS Relay debugging function. To enable DNS Relay debugging, use the command:

```
awplus# debug ip dns forwarding
```

To check the status of DNS Relay, use the command:

```
awplus# show ip dns forwarding
```

To display the DNS Relay name resolver cache, use the command:

```
awplus# show ip dns forwarding cache
```

DHCP options

When your device is using its DHCP client for an interface, it can receive the following DHCP options from the DHCP server:

- Option 6 - a list of DNS servers. This list appends to the DNS servers set on your device with the ip name-server command.
- Option 15 - a domain name used to resolve host names. This option replaces the domain name set with the ip domain-name command.

See Chapter 60, Dynamic Host Configuration Protocol (DHCP) Introduction for more information about DHCP and DHCP options.

Internet Control Message Protocol (ICMP)

The Internet Control Message Protocol (ICMP) allows networking devices to send information and control messages to other devices or hosts. Your device implements all non-obsolete ICMP functions.

The following table lists the ICMP messages implemented by your device.

ICMP Message Type	Device Response
Echo reply (0)	This is used to implement the ping command. Your device sends out an echo reply in response to an echo request.
Destination unreachable (3)	This message is sent when your device drops a packet because it did not have a route to the destination.
Redirect (5)	Your device issues this message to inform a local host that its target is located on the same LAN (no routing is required) or when it detects a host using a non-optimal route (usually because a link has failed or changed its status). For example, if your device receives a packet destined to its own MAC address, but with a destination IP address of another host in the local subnet, it returns an ICMP redirect to the originating host. ICMP redirects are disabled on interfaces on which local proxy ARP is enabled.
Echo request (8)	This is related to echo replies. If your device receives an echo request, it sends an echo reply. If you enter the ping command, your device generates echo requests.
Router Advertisements (10)	These are Router Discovery Protocol messages. If Router Discovery is enabled, your device sends these to announce the IP addresses of the sending interface.
Time to Live Exceeded (11)	If the TTL field in a packet falls to zero, your device sends this message.This occurs when there are too many hops in the path that a packet is traversing.

ICMP messages are enabled on all interfaces by default. You can control the flow of ICMP messages across different interfaces using the **access-list** commands. See Chapter 32, IPv4 Hardware Access Control List (ACL) Commands and Chapter 33, IPv4 Software Access Control List (ACL) Commands.

ICMP Router Discovery Protocol (IRDP)

Router discovery

Your device supports the router specification sections of RFC 1256, ***ICMP Router Discovery Messages***. If this feature is configured, your device sends router advertisements periodically and in response to router solicitations. It does not support the Host Specification section of this RFC.

Benefits

Before an IP host can send an IP packet, the host has to know the IP address of a neighboring router that can forward the packet to its destination. ICMP Router Discovery messages let routers automatically advertise themselves to hosts. Other methods either require someone to manually keep these addresses current, or require DHCP to send router addresses.

Router discovery process

The following table summarizes what happens when Router Discovery advertisements are enabled on an interface.

When...	**Then...**
Router Discovery advertising starts on an interface because: ■ your device starts up, or ■ you enable advertisements on your device or on an interface	your device multicasts a router advertisement and continues to multicast them periodically until router advertising is disabled.
a host starts up	the host may send a router solicitation message.
your device receives a router solicitation	your device multicasts an early router advertisement from the interface on which it received the router solicitation.
a host receives a router advertisement	the host stores the IP address and preference level for the advertisement lifetime.
the lifetimes of all existing router advertisements on a host expire	the host sends a router solicitation.
a host does not receive a router advertisement after sending a small number of router solicitations	the host waits for the next unsolicited router advertisement.
a host needs a default router address	the host uses the IP address of the router or L3 switch with the highest preference level.
Router Discovery advertising is deleted from the interface	your device multicasts a router advertisement with the IP address(es) that stopped advertising, and a lifetime of zero. It continues to periodically multicast router advertisements for other interfaces, if configured to.
the router receives a router advertisement from another router	the router does nothing but silently discards the message

Advertisement messages

A router advertisement is an ICMP (type 10) message that contains the following:

- in the destination address field of the IP header, the interface's configured advertisement address, either 224.0.0.1 or 255.255.255.255.
- in the lifetime field, the interface's configured advertisement lifetime.
- in the Router Address and Preference Level fields, the addresses and preference levels of all the logical interfaces that are set to advertise.

Your device does not send router advertisements by default.

Solicitation message

A router solicitation is an ICMP (type 10) message containing:

- source Address: an IP address belonging to the interface from which the message is sent
- destination Address: the configured Solicitation Address, and
- Time-to-Live: 1 if the Destination Address is an IP multicast address; at least 1 otherwise.

Advertisement interval

The router advertisement interval is the time between router advertisements. For the first few advertisements sent from an interface (up to 3), your device sends the router advertisements at intervals of at most 16 seconds. After these initial transmissions, it sends router advertisements at random intervals between the minimum and maximum intervals that the user configures, to reduce the probability of synchronization with the advertisements from other routers on the same link. By default, the minimum is 450 seconds (7.5 minutes), and the maximum is 600 seconds (10 minutes).

Preference level

The preference level is the preference of the advertised address as a default router address relative to other router addresses on the same subnet. By default, all routers and Layer 3 switches have the same preference level, zero. While it is entered as a decimal from 0 to 2147483647, it is encoded in router advertisements as a twos-complement hex integer from 0x8000000 to 0x7fffffff. A higher preference level is preferred over a lower value.

Lifetime

The lifetime of a router advertisement is how long the information in the advertisement is valid. By default, the lifetime of all advertisements is 1800 seconds (30 minutes).

Address type

Your device can send its router advertisements using either a broadcast or multicast destination address. By default, your device sends router advertisements using the all-systems multicast address (224.0.0.1). However, on networks where the hosts do not support IP multicast you must use the broadcast address (255.255.255.255). To change the address type to broadcast on an interface, use the command:

```
awplus(config-if)# ip irdp broadcast
```

To change the address type back to multicast, use the **no** variant of the above command, or use the command:

```
awplus(config-if)# ip irdp multicast
```

Configuration procedure

Do the following to configure your device to send router advertisements.

Step 1: Enter the interface to advertise.

Enter the configuration mode for the interface, using the command:

```
awplus(config)# interface <interface>
```

Step 2: Change the address type.

By default, your device sends router advertisements using a multicast destination address. If hosts on your network do not support this, change the address type to broadcast, using the command:

```
awplus(config-if)# ip irdp broadcast
```

Step 3: Configure the advertisement interval and lifetime.

By default, your device sends router advertisements every 7.5 to 10 minutes, with a lifetime of 30 minutes. These settings are likely to work well in most situations, and will not cause a large amount of extra traffic, even if there are several routers on the LAN. If you change these settings, keep the following proportions:

```
lifetime=3 x maxadvertisementinterval
minadvertisementinverval=0.75 x maxadvertisementinterval
```

You cannot set the maximum advertisement interval below the minimum interval. If you are lowering the maximum interval to a value below the current minimum interval, you must change the minimum value first. This also applies to changing the minimum interval above the current maximum interval.

To change the maximum advertisement interval, use the command:

```
awplus(config-if)# ip irdp maxadvertinterval <4-1800>
```

To change the minimum advertisement interval, use the command:

```
awplus(config-if)# ip irdp minadvertinterval <3-1800>
```

To change the lifetime for your device's router advertisements, use the command:

```
awplus(config-if)# ip irdp lifetime <0-9000>
```

Step 4: Set preference levels.

By default, every interface has the same preference for becoming a default router. To give the interface a higher preference, increase the preference level. To give it a lower preference, decrease this value.

To set the preference level for all addresses on this interface, use the command:

```
awplus(config-if)# ip irdp preference <0-2147483647>
```

To set the preference for a specific address on the interface, use the command:

```
awplus(config-if)# ip irdp address <ip-address> preference
                   <0-2147483647>
```

Step 5: Enable advertising on the interface.

To enable router advertisements on an interface, enter the interface mode and use the command:

```
awplus(config-if)# ip irdp
```

Step 6: Enable advertising on your device.

To globally enable router advertisements on your device, enter the configure mode and use the command:

```
awplus(config)# router ip irdp
```

Step 7: Check advertise settings.

To view the IRDP configuration on the interface, use the command:

```
awplus# show ip irdp interface [<interface-name>]
```

To view the global IRDP configuration for your device, use the command:

```
awplus# show ip irdp
```

Debugging IRDP

Information which may be useful for troubleshooting IRDP is available using the IRDP debugging function. To enable IRDP debugging, use the command:

```
awplus# debug ip irdp {event|nsm|receive|send|both|
        detail|all}
```

Checking IP Connections

To verify connections between networks and network devices, use the ping (Packet Internet Groper) and trace route functions on your device.

Ping

Ping tests the connectivity between two network devices to determine whether each network device can "see" the other device. Echo request packets are sent to the destination addresses and responses are displayed on the console.

If you can ping the end destination, then the physical, Layer 2 and Layer 3 links are functioning, and any difficulties are in the network or higher layers.

If pinging the end destination fails, use traceroute to discover the point of failure in the route to the destination.

To ping a device, use the command:

```
awplus# ping {<hostname>|<ipaddr>}
```

Traceroute

You can use traceroute to discover the route that packets pass between two systems running the IP protocol. Traceroute sends an initial UDP packets with the Time To Live (TTL) field in the IP header set starting at 1. The TTL field is increased by one for every subsequent packet sent until the destination is reached. Each hop along the path between two systems responds with a TTL exceeded packet (ICMP type 11) and from this the path is determined.

To use traceroute, use the command:

Enter either the hostname or the IP address of the device you are trying to reach.

IP Helper

The IP Helper feature allows the switch to receive UDP broadcasts on one subnet, and forward them as broadcasts or unicasts into another subnet, so a client can use an application which uses UDP broadcast (such as Net-BIOS) when the client and server are located in different subnets. The IP Helper feature forwards UDP broadcast network traffic to specific hosts on another subnet and/or to the broadcast address of another subnet.

When the IP Helper feature is enabled on a VLAN interface, the UDP broadcast packets received on the interface are processed for forwarding out through another interface into another subnet. Depending on the nature of the ip-helper addresses configured, the UDP broadcasts will be unicast forwarded to a single host in the destination subnet, or unicast forwarded to multiple hosts in the destination subnet, or broadcast to the broadcast address of the destination subnet. Not all UDP broadcasts will be forwarded when IP Helper is configured. The set of broadcasts to be forwarded can be defined by specifying the destination UDP port(s) of the packets you wish to forward.

The command to enable the forwarding of UDP broadcasts received on a given interface is ip helper-address (entered in interface configuration mode). The ip forward-protocol udp command specifies types of broadcast packets to forward.

Multiple different destination addresses can be specified by using multiple instances of the ip helper-address command under the same interface. If a destination address is specified that is actually the broadcast address of one of the subnets directly connected to the switch, then the UDP packets will be forwarded as broadcasts onto that subnet.

Likewise, multiple different types of UDP packet can be specified for forwarding by specifying multiple different destination ports using the ip forward-protocol udp command.

Note The types of UDP broadcast packets that the switch will forward are **only** those specified by the **ip forward-protocol** command(s). There are not other UDP packet types that the IP helper process forwards by default.

IP Directed Broadcast

IP directed-broadcast is enabled and disabled per VLAN interface. When enabled a directed broadcast packet is forwarded to an enabled VLAN interface if received on another subnet.

An IP directed broadcast is an IP packet whose destination address is a broadcast address for some IP subnet, but originates from a node that is not itself part of that destination subnet. When a directed broadcast packet reaches a switch that is directly connected to its destination subnet, the packet is flooded as a broadcast on the destination subnet.

The ip directed-broadcast command controls the flooding of directed broadcasts when they reach target subnets. The command affects the final transmission of the directed broadcast on its destination subnet. It does not affect the transit unicast routing of IP directed broadcasts. If directed broadcast is enabled for an interface, incoming directed broadcast IP packets intended for the subnet assigned to interface will be flooded as broadcasts on that subnet.

If the **no ip directed-broadcast** command is configured for an interface, directed broadcasts destined for the subnet where the interface is attached will be dropped instead of broadcast.

Chapter 25: IP Addressing and Protocol Commands

Introduction

This chapter provides an alphabetical reference of commands used to configure the following protocols:

- Address Resolution Protocol (ARP)
- Domain Name Service (DNS)
- ICMP Router Discovery Advertisements (IRDP)

For more information see Chapter 24, Internet Protocol (IP) Addressing and Protocols.

Command List

arp-aging-timeout

This command sets a timeout period on dynamic ARP entries associated with a specific interface. If your device stops receiving traffic for the host specified in a dynamic ARP entry, it deletes the ARP entry from the ARP cache after this timeout is reached.

Your device times out dynamic ARP entries to ensure that the cache does not fill with entries for hosts that are no longer active. Static ARP entries are not aged or automatically deleted.

By default the time limit for dynamic ARP entries is 300 seconds on all interfaces.

The **no** variant of this command sets the time limit to the default of 300 seconds.

Syntax

```
arp-aging-timeout <0-432000>

no arp-aging timeout
```

Parameter	Description
<0-432000>	The timeout period in seconds.

Default 300 seconds (5 minutes)

Mode Interface Configuration for a VLAN interface.

Example To set the ARP entries on interface `vlan30` to time out after two minutes, use the commands:

```
    awplus(config)# interface vlan30
 awplus(config-if)# arp-aging-timeout 120
```

Related Commands clear arp-cache
show arp

arp (IP address MAC address)

This command adds a static ARP entry to the ARP cache. This is typically used to add entries for hosts that do not support ARP or to speed up the address resolution function for a host. The ARP entry must not already exist. Use the **alias** parameter to allow your device to respond to ARP requests for this IP address.

The **no** variant of this command removes the static ARP entry. Use the clear arp-cache command on page 25.9 to remove the dynamic ARP entries in the ARP cache.

Syntax

```
arp <ip-addr> <mac-address> [<port-number>] [alias]
```

```
no arp <ip-addr>
```

Parameter	Description
<ip-addr>	IPv4 address of the device you are adding as a static ARP entry.
<mac-address>	MAC address of the device you are adding as a static ARP entry, in hexadecimal notation with the format HHHH.HHHH.HHHH.
<port-number>	The port number associated with the IP address. Specify this when the IP address is part of a VLAN.
alias	Allows your device to respond to ARP requests for the IP address. Proxy ARP must be enabled on the interface before using this parameter.

Mode Global Configuration

Example To add the IP address 10.10.10.9 with the MAC address 0010.2533.4655 into the ARP cache, and have your device respond to ARP requests for this address, use the commands:

```
        awplus# configure terminal
awplus(config)# arp 10.10.10.9 0010.2355.4566 alias
```

Related Commands

clear arp-cache
ip proxy-arp
show arp

arp log

This command enables the logging of dynamic and static ARP entries in the ARP cache. The ARP cache contains mappings of switch ports, VLAN IDs, and IP addresses to physical MAC addresses for hosts.

This command can display the MAC addresses in the ARP log either using the default hexadecimal notation (HHHH.HHHH.HHHH), or using the IEEE standard hexadecimal notation (HH-HH-HH-HH-HH-HH).

Use the **no** variant of this command to disable the logging of dynamic and static ARP entries in the ARP cache.

Syntax

```
arp log [mac-address-format ieee]
```

```
no arp log [mac-address-format ieee]
```

Parameter	Description
`mac-address-format ieee`	Display the MAC address in hexadecimal notation with the standard IEEE format (HH-HH-HH-HH-HH-HH), instead of displaying the MAC address with the default hexadecimal format (HHHH.HHHH.HHHH).

Default The ARP logging feature is disabled by default.

Mode Global Configuration

Usage You have the option to change how the MAC address is displayed in the ARP log message, to use the default hexadecimal notation (HHHH.HHHH.HHHH), or the IEEE format hexadecimal notation (HH-HH-HH-HH-HH-HH) when you apply the **mac-address-format ieee** parameter.

Enter the **arp log** command without the optional **mac-address-format ieee** parameter specified for MAC addresses in the ARP log output to use the default hexadecimal notation (HHHH.HHHH.HHHH).

Enter the **arp log mac-address-format ieee** command for MAC addresses in the ARP log output to use the IEEE standard format hexadecimal notation (HH-HH-HH-HH-HH-HH).

Use the **no** variant of this command (**no arp log**) without the optional **mac-address-format ieee** parameter specified to disable ARP logging on the switch

Use the **no** variant of this command with the optional **mac-address-format ieee** parameter specified (**no arp log mac-address-format ieee**) to disable IEEE standard format hexadecimal notation (HH-HH-HH-HH-HH-HH) and revert to the default hexadecimal notation (HHHH.HHHH.HHHH) for MAC addresses in the ARP log output.

To display ARP log messages use the **show log | include ARP_LOG** command.

Examples To enable ARP logging and use the default hexadecimal notation (HHHH.HHHH.HHHH), use the following commands:

```
        awplus# configure terminal
awplus(config)# arp log
```

To disable ARP logging on the switch of MAC addresses displayed using the default hexadecimal notation (HHHH.HHHH.HHHH), use the following commands:

```
        awplus# configure terminal
awplus(config)# no arp log
```

To enable ARP logging and to specify that the MAC address in the log message is displayed in the standard IEEE format hexadecimal notation (HH-HH-HH-HH-HH-HH), use the following commands:

```
        awplus# configure terminal
awplus(config)# arp log mac-address-format ieee
```

To disable ARP logging on the switch of MAC addresses displayed using the standard IEEE format hexadecimal notation (HH-HH-HH-HH-HH-HH), and revert to the use of the default hexadecimal notation (HHHH.HHHH.HHHH) instead, use the following commands:

```
        awplus# configure terminal
awplus(config)# no arp log mac-address-format ieee
```

To display ARP log messages, use following command:

```
        awplus# show log | include ARP_LOG
```

Output Below is example output from the **show log | include ARP_LOG** command after enabling ARP logging displaying default hexadecimal notation MAC addresses (HHHH.HHHH.HHHH) using the **arp log** command.

Figure 25-1: Example output from the **show log | include ARP_LOG** command

```
awplus#configure terminal
awplus(config)#arp log
awplus(config)#exit
awplus#show log | include ARP_LOG
2010 Apr  6 06:21:01 user.notice awplus HSL[1007]: ARP_LOG port1.0.7 vlan1 add
0013.4078.3b98 (192.168.2.4)
2010 Apr  6 06:22:30 user.notice awplus HSL[1007]: ARP_LOG port1.0.7 vlan1 del
0013.4078.3b98 (192.168.2.4)
2010 Apr  6 06:23:26 user.notice awplus HSL[1007]: ARP_LOG port1.0.7 vlan1 add
0030.940e.136b (192.168.2.20)
2010 Apr  6 06:23:30 user.notice awplus IMISH[1830]: show log | include ARP_LOG
```

Below is example output from the **show log | include ARP_LOG** command after enabling ARP logging displaying IEEE standard format hexadecimal notation MAC addresses (HH-HH-HH-HH-HH-HH) using the **arp log mac-address format ieee** command.

Figure 25-2: Example output from the **show log | include ARP_LOG** command

```
awplus#configure terminal
awplus(config)#arp log mac-address-format ieee
awplus(config)#exit
awplus#show log | include ARP_LOG
2010 Apr  6 06:25:28 user.notice awplus HSL[1007]: ARP_LOG port1.0.7 vlan1 add 00-
17-9a-b6-03-69 (192.168.2.12)
2010 Apr  6 06:25:30 user.notice awplus HSL[1007]: ARP_LOG port1.0.7 vlan1 add 00-
03-37-6b-a6-a5 (192.168.2.10)
2010 Apr  6 06:26:53 user.notice awplus HSL[1007]: ARP_LOG port1.0.7 vlan1 del 00-
30-94-0e-13-6b (192.168.2.20)
2010 Apr  6 06:27:31 user.notice awplus HSL[1007]: ARP_LOG port1.0.7 vlan1 del 00-
17-9a-b6-03-69 (192.168.2.12)
2010 Apr  6 06:28:09 user.notice awplus HSL[1007]: ARP_LOG port1.0.7 vlan1 del 00-
03-37-6b-a6-a5 (192.168.2.10)
2010 Apr  6 06:28:14 user.notice awplus IMISH[1830]: show log | include ARP_LOG
```

Below are the parameters in output of the **show log | include ARP_LOG** command with an ARP log message format of <*ARP_LOG*> <*port number*> <*VLAN ID*> <*Operation*> <*MAC*> <*IP*> after <*date*> <*time*> <*severity*> <*hostname*> <*program name*> information.

Table 25-1: Parameters in output of the **show log | include ARP_LOG** command

Parameter	Description
<*ARP_LOG*>	Indicates ARP log entry information follows <*date*> <*time*> <*severity*> <*hostname*> <*program name*> log information.
<*port number*>	Indicates switch port number for the ARP log entry.
<*VLAN ID*>	Indicates the VLAN ID for the ARP log entry.
<*Operation*>	Indicates 'add' if the ARP log entry displays an ARP addition. Indicates 'del' if the ARP log entry displays an ARP deletion.
<*MAC*>	Indicates the MAC address for the ARP log entry, either in the default hexadecimal notation (HHHH.HHHH.HHHH) or in the IEEE standard format hexadecimal notation (HH-HH-HH-HH-HH-HH) as specified with the **arp log** or the **arp log mac-address-format ieee** command.
<*IP*>	Indicates the IP address for the ARP log entry.

Validation Commands show running-config

Related Commands show log

arp opportunistic-nd

Use this command to enable opportunistic neighbor discovery for the global ARP cache. Opportunistic neighbor discovery changes the behavior for unsolicited ARP packet forwarding on the switch.

Use the **no** variant of this command to disable opportunistic neighbor discovery for the global ARP cache.

Syntax

```
arp opportunistic-nd

no arp opportunistic-nd
```

Default Opportunistic neighbor discovery is disabled by default.

Mode Global Configuration

Usage When opportunistic neighbor discovery is enabled, the switch will reply to any received unsolicited ARP packets (but not gratuitous ARP packets). The source MAC address for the unsolicited ARP packet is added to the ARP cache, so the switch forwards the ARP packet. When opportunistic neighbor discovery is disabled, the source MAC address for the ARP packet is not added to the ARP cache, so the ARP packet is not forwarded by the switch.

Note this command enables or disables opportunistic neighbor discovery for a VRF instance if the **vrf** parameter and an instance name are applied. If a VRF instance is not specified, then opportunistic neighbor discovery is enabled or disabled for switch ports configured for IPv4.

Use the ipv6 opportunistic-nd command to enable optimistic neighbor discovery in Global Configuration mode for all interfaces on the switch configured for IPv6. Use a show arp command to confirm opportunistic neighbor discovery is configured on the switch.

Example To enable opportunistic neighbor discovery for the global ARP cache, enter:

```
        awplus#  configure terminal
awplus(config)#  arp opportunistic-nd
```

To disable opportunistic neighbor discovery for the global ARP cache, enter:

```
        awplus#  configure terminal
awplus(config)#  no arp opportunistic-nd
```

Related Commands ipv6 opportunistic-nd

Validation Commands show arp

clear arp-cache

This command deletes dynamic ARP entries from the ARP cache.

Syntax x600

```
clear arp-cache
```

Mode Privileged Exec

Usage To display the entries in the ARP cache, use the show arp command. To remove static ARP entries, use the no variant of the arp (IP address MAC address) command on page 25.4.

Examples To clear all dynamic ARP entries, use the command:

```
awplus# clear arp-cache
```

Related Commands

arp-aging-timeout
arp (IP address MAC address)
show arp

clear ip dns forwarding cache

Use this command to clear the DNS Relay name resolver cache.

Syntax

```
clear ip dns forwarding cache
```

Mode Privileged Exec

Examples To clear the DNS Relay name resolver cache, use the command:

```
awplus# clear ip dns forwarding cache
```

Related Commands ip dns forwarding cache

debug ip dns forwarding

Use this command to enable DNS Relay debugging.

Use the **no** variant of this command to disable DNS Relay debugging.

Syntax

```
debug ip dns forwarding
```

```
no debug ip dns forwarding
```

Default DNS Relay debugging is disabled by default.

Mode Privileged Exec

Examples To enable DNS forwarding debugging, use the commands:

```
awplus# debug ip dns forwarding
```

To disable DNS forwarding debugging, use the commands:

```
awplus# no debug ip dns forwarding
```

Related Commands ip dns forwarding
show debugging ip dns forwarding

debug ip packet interface

The **debug ip packet interface** command enables IP packet debug and is controlled by the **terminal monitor** command.

If the optional **icmp** keyword is specified then ICMP packets are shown in the output.

The **no** variant of this command disables the **debug ip interface** command.

Syntax

```
debug ip packet interface {<interface-name>|all}
    [address <ip-address>|verbose|hex|arp|udp|tcp|icmp]

no debug ip packet interface [<interface-name>]
```

Parameter	Description
`<interface>`	Specify a single Layer 3 interface name (not a range of interfaces) This keyword can be specified as either all or as a single Layer 3 interface to show debugging for either all interfaces or a single interface.
`all`	Specify all Layer 3 interfaces on the switch.
`<ip-address>`	Specify an IPv4 address. If this keyword is specified, then only packets with the specified IP address as specified in the ip-address placeholder are shown in the output.
`verbose`	Specify **verbose** to output more of the IP packet. If this keyword is specified then more of the packet is shown in the output.
`hex`	Specify **hex** to output the IP packet in hexadecimal. If this keyword is specified, then the output for the packet is shown in hex.
`arp`	Specify **arp** to output ARP protocol packets. If this keyword is specified, then ARP packets are shown in the output.
`udp`	Specify **udp** to output UDP protocol packets. If this keyword is specified then UDP packets are shown in the output.
`tcp`	Specify **tcp** to output TCP protocol packets. If this keyword is specified, then TCP packets are shown in the output.
`icmp`	Specify **icmp** to output ICMP protocol packets. If this keyword is specified, then ICMP packets are shown in the output.

Mode Privileged Exec and Global Configuration

Examples To turn on ARP packet debugging on `vlan1`, use the command:

```
awplus# debug ip packet interface vlan1 arp
```

To turn on all packet debugging on all interfaces on the switch, use the command:

```
awplus# debug ip packet interface all
```

To turn on TCP packet debugging on `vlan1` and IP address `192.168.2.4` , use the command:

```
awplus# debug ip packet interface vlan1 address 192.168.2.4
        tcp
```

To turn off IP packet interface debugging on all interfaces, use the command:

```
awplus# no debug ip packet interface
```

To turn off IP packet interface debugging on interface `vlan2`, use the command:

```
awplus# no debug ip packet interface vlan2
```

Related Commands

no debug all
show debugging ip dns forwarding
tcpdump
terminal monitor
undebug ip packet interface

debug ip irdp

This command enables debugging of ICMP Router Discovery Protocol (IRDP) events and messages on your device. IRDP debugging is disabled by default.

The **no** variant of this command disables IRDP debugging. Negating any packet debug mode will switch detail off.

Syntax

```
debug ip irdp {event|nsm|receive|send|both|detail|all}
```

```
no debug ip irdp {event|nsm|receive|send|both|detail|all}
```

Parameter	Description
`event`	Enables debugging of IRDP events.
`nsm`	Enables debugging of IRDP processing of NSM messages.
`receive`	Enables debugging of IRDP input packet processing.
`send`	Enables debugging of IRDP output packet processing.
`both`	Enables debugging of both IRDP input and output packet processing.
`detail`	Enables detailed debugging of both IRDP input and output packet processing. Note that setting detail also sets both, so if you set **detail**, the output will show "packet debugging mode is all". Negating any packet debug mode will switch detail off.
`all`	Enables all IRDP debugging types.

Default IRDP protocol debugging is disabled by default.

Mode Privileged Exec and Global Configuration

Examples To enable IRDP input packet process debugging, use the following command:

```
awplus# debug ip irdp receive
```

To disable all IRDP debugging, use the following command:

```
awplus# no debug ip irdp all
```

Related Commands

ip irdp
router ip irdp
show ip irdp
undebug ip irdp

ip address

This command sets a static IP address on an interface. To set the primary IP address on the interface, specify only **ip address** *<ip-address/m>*. This overwrites any configured primary IP address. To add additional IP addresses on this interface, use the **secondary** parameter. You must configure a primary address on the interface before configuring a secondary address.

Note Use **show running-config** interface not **show ip interface brief** when you need to view a secondary address configured on an interface. **show ip interface brief** will only show the primary address not a secondary address for an interface.

The **no** variant of this command removes the IP address from the interface. You cannot remove the primary address when a secondary address is present.

Syntax

```
ip address <ip-addr/prefix-length> [secondary [label <label>]]

no ip address <ip-addr/prefix-length> [secondary]

no ip address
```

Parameter	Description
<ip-addr/prefix-length>	The IPv4 address and prefix length you are assigning to the interface.
`label`	Adds a user-defined description of the secondary IP address.
<label>	A user-defined description of the secondary IP address. Valid characters are any printable character and spaces.

Mode Interface Configuration for a VLAN interface or a local loopback interface.

Examples To add the primary IP address `10.10.10.50/24` to the interface `vlan3`, use the following commands:

```
          awplus# configure terminal
  awplus(config)# interface vlan3
awplus(config-if)# ip address 10.10.10.50/24
```

To add the secondary IP address `10.10.11.50/24` to the same interface, use the following commands:

```
          awplus# configure terminal
  awplus(config)# interface vlan3
awplus(config-if)# ip address 10.10.11.50/24 secondary
```

To add the IP address 10.10.11.50/24 to the local loopback interface lo, use the following commands:

```
          awplus# configure terminal
  awplus(config)# interface lo
awplus(config-if)# ip address 10.10.11.50/24
```

Related Commands interface (to configure)
show ip interface
show running-config interface

ip dns forwarding

Use this command to enable DNS Relay, the forwarding of incoming DNS queries for IP hostname-to-address translation.

Use the **no** variant of this command to disable the forwarding of incoming DNS queries for IP hostname-to-address translation.

Syntax

```
ip dns forwarding

no ip dns forwarding
```

Default The forwarding of incoming DNS query packets is disabled by default.

Mode Global Configuration

Usage See "DNS Relay" on page 24.10 for more information about DNS Relay to map IPv4 and IPv6 addresses to name servers to maintain a a database of hostname-to-address mappings.

Examples To enable the forwarding of incoming DNS query packets, use the commands:

```
        awplus# configure terminal
awplus(config)# ip dns forwarding
```

To disable the forwarding of incoming DNS query packets, use the commands:

```
        awplus# configure terminal
awplus(config)# no ip dns forwarding
```

Related Commands debug ip dns forwarding
ip dns forwarding cache
ip dns forwarding retry
ip dns forwarding source-interface
ip dns forwarding timeout
ip name-server
show ip dns forwarding

ip dns forwarding cache

Use this command to set the DNS Relay name resolver cache size and cache entry lifetime period. The DNS Relay name resolver cache stores the mappings between domain names and IP addresses.

Use the **no** variant of this command to set the default DNS Relay name resolver cache size and cache entry lifetime period.

Syntax

```
ip dns forwarding cache [size <0-1000>] [timeout <60-3600>]

no ip dns forwarding cache [size|timeout]
```

Parameter	Description
<0-1000>	Number of entries in the DNS Relay name resolver cache.
<60-3600>	Timeout value in seconds.

Default The default cache size is 0 (no entries) and the default lifetime is 1800 seconds.

Mode Global Configuration

Usage See "DNS Relay" on page 24.10 for more information about DNS Relay to map IPv4 and IPv6 addresses to name servers to maintain a a database of hostname-to-address mappings.

Examples To set the cache size to 10 entries and the lifetime to 500 seconds, use the commands:

```
        awplus# configure terminal
awplus(config)# ip dns forwarding cache size 10 time 500
```

To set the cache size to the default, use the commands:

```
        awplus# configure terminal
awplus(config)# no ip dns forwarding cache size
```

Related Commands

clear ip dns forwarding cache
ip dns forwarding
ip dns forwarding retry
ip dns forwarding source-interface
ip dns forwarding timeout
show ip dns forwarding cache

ip dns forwarding retry

Use this command to set the number of times DNS Relay will retry to forward DNS queries.

Use the **no** variant of this command to set the number of retries to the default of 2.

Syntax

```
ip dns forwarding retry <0-100>
```

```
no ip dns forwarding retry
```

Parameter	Description
<0-100>	Number of times DNS Relay will retry to forward a DNS query.

Default The default number of retries is 2.

Mode Global Configuration

Usage See “DNS Relay” on page 24.10 for more information about DNS Relay to map IPv4 and IPv6 addresses to name servers to maintain a a database of hostname-to-address mappings.

Examples To set the retry count to 9, use the commands:

```
         awplus# configure terminal
 awplus(config)# ip dns forwarding retry 9
```

To set the retry count to the default of 2, use the commands:

```
         awplus# configure terminal
 awplus(config)# no ip dns forwarding retry
```

Related Commands

ip dns forwarding
ip dns forwarding cache
ip dns forwarding source-interface
ip dns forwarding timeout

ip dns forwarding source-interface

Use this command to set the interface to use for forwarding and receiving DNS queries.

Use the **no** variant of this command to unset the interface used for forwarding and receiving DNS queries.

Syntax

```
ip dns forwarding source-interface <interface-name>
```

```
no ip dns forwarding source-interface
```

Parameter	Description
<interface-name>	An alphanumeric string that is the interface name.

Default The default is that no interface is set and the switch selects the appropriate source IP address automatically.

Mode Global Configuration

Usage See "DNS Relay" on page 24.10 for more information about DNS Relay to map IPv4 and IPv6 addresses to name servers to maintain a a database of hostname-to-address mappings.

Examples To set `vlan1` as the source interface for relayed DNS queries, use the commands:

```
        awplus# configure terminal
awplus(config)# ip dns forwarding source-interface vlan1
```

To clear the source interface for relayed DNS queries, use the commands:

```
        awplus# configure terminal
awplus(config)# no ip dns forwarding source-interface
```

Related Commands ip dns forwarding
ip dns forwarding cache
ip dns forwarding retry
ip dns forwarding timeout

ip dns forwarding timeout

Use this command to set the time period for the DNS Relay to wait for a DNS response.

Use the **no** variant of this command to set the time period to wait for a DNS response to the default of 3 seconds.

Syntax

```
ip dns forwarding timeout <0-3600>
```

```
no ip dns forwarding timeout
```

Parameter	Description
<0-3600>	Timeout value in seconds.

Default The default timeout value is 3 seconds.

Mode Global Configuration

Usage See "DNS Relay" on page 24.10 for more information about DNS Relay to map IPv4 and IPv6 addresses to name servers to maintain a a database of hostname-to-address mappings.

Examples To set the timeout value to 12 seconds, use the commands:

```
        awplus# configure terminal
awplus(config)# ip dns forwarding timeout 12
```

To set the timeout value to the default of 3 seconds, use the commands:

```
        awplus# configure terminal
awplus(config)# no ip dns forwarding timeout
```

Related Commands ip dns forwarding
ip dns forwarding cache
ip dns forwarding retry
ip dns forwarding source-interface

ip domain-list

This command adds a domain to the DNS list. Domain are appended to incomplete host names in DNS requests. Each domain in this list is tried in turn in DNS lookups. This list is ordered so that the first entry you create is checked first.

The **no** variant of this command deletes a domain from the list.

Syntax

```
ip domain-list <domain-name>

no ip domain-list <domain-name>
```

Parameter	Description
<domain-name>	Domain string, for example "company.com".

Mode Global Configuration

Usage If there are no domains in the DNS list, then your device uses the domain specified with the ip domain-name command. If any domain exists in the DNS list, then the device does not use the domain set using the **ip domain-name** command.

See "Domain Name System (DNS)" on page 24.8 for introductory information about DNS. See "DNS Client" on page 24.9 for information about DNS Client configuration commands.

Example To add the domain `example.net` to the DNS list, use the following commands:

```
        awplus# configure terminal

awplus(config)# ip domain-list example.net
```

Related Commands

ip domain-lookup
ip domain-name
show ip domain-list

ip domain-lookup

This command enables the DNS client on your device. This allows you to use domain names instead of IP addresses in commands. The DNS client resolves the domain name into an IP address by sending a DNS enquiry to a DNS server, specified with the ip name-server command.

The **no** variant of this command disables the DNS client. The client will not attempt to resolve domain names. You must use IP addresses to specify hosts in commands.

Syntax

```
ip domain-lookup

no ip domain-lookup
```

Mode Global Configuration

Usage The client is enabled by default. However, it does not attempt DNS enquiries unless there is a DNS server configured.

See "DNS Client" on page 24.9 for information about DNS Client configuration commands.

Examples To enable the DNS client on your device, use the following commands:

```
        awplus#  configure terminal
  awplus(config)#  ip domain-lookup
```

To disable the DNS client on your device, use the following commands:

```
        awplus#  configure terminal
  awplus(config)#  no ip domain-lookup
```

Related Commands

ip domain-list
ip domain-name
ip name-server
show hosts
show ip name-server

ip domain-name

This command sets a default domain for the DNS. The DNS client appends this domain to incomplete host-names in DNS requests.

The **no** variant of this command removes the domain-name previously set by this command.

Syntax

```
ip domain-name <domain-name>

no ip domain-name <domain-name>
```

Parameter	Description
<domain-name>	Domain string, for example "company.com".

Mode Global Configuration

Usage If there are no domains in the DNS list (created using the ip domain-list command) then your device uses the domain specified with this command. If any domain exists in the DNS list, then the device does not use the domain configured with this command.

See "DNS Client" on page 24.9 for information about DNS Client configuration commands.

When your device is using its DHCP client for an interface, it can receive Option 15 from the DHCP server. This option replaces the domain name set with this command. See Chapter 60, Dynamic Host Configuration Protocol (DHCP) Introduction for more information about DHCP and DHCP options.

Example To configure the domain name, enter the following commands:

```
        awplus# configure terminal
awplus(config)# ip domain-name company.com
```

Related Commands

ip domain-list
show ip domain-list
show ip domain-name

ip directed-broadcast

Use this command to enable flooding of directed broadcast packets into a directly connected subnet. If this command is configured on a VLAN interface, then directed broadcasts received on other VLAN interfaces, destined for the subnet on this VLAN, will be flooded to the subnet broadcast address of this VLAN.

Use the **no** variant of this command to disable **ip directed-broadcast**. When this feature is disabled using the **no** variant of this command, directed broadcasts are not forwarded.

Syntax

```
ip directed-broadcast
```

```
no ip directed-broadcast
```

Default The **ip directed-broadcast** command is disabled by default.

Mode Interface Configuration for a VLAN interface.

Usage IP directed-broadcast is enabled and disabled per VLAN interface. When enabled a directed broadcast packet is forwarded to an enabled VLAN interface if received on another subnet.

An IP directed broadcast is an IP packet whose destination address is a broadcast address for some IP subnet, but originates from a node that is not itself part of that destination subnet. When a directed broadcast packet reaches a switch that is directly connected to its destination subnet, that packet is flooded as a broadcast on the destination subnet.

The **ip directed-broadcast c**ommand controls the flooding of directed broadcasts when they reach target subnets. The command affects the final transmission of the directed broadcast on its destination subnet. It does not affect the transit unicast routing of IP directed broadcasts. If directed broadcast is enabled for an interface, incoming directed broadcast IP packets intended for the subnet assigned to interface will be flooded as broadcasts on that subnet.

If the **no ip directed-broadcast** command is configured for an interface, directed broadcasts destined for the subnet where the interface is attached will be dropped instead of broadcast.

Examples To enable **ip directed-broadcast**, to flood broadcast packets out via the `vlan2` interface, enter the following commands:

```
           awplus# configure terminal
   awplus(config)# interface vlan2
awplus(config-if)# ip directed-broadcast
```

To disable **ip directed-broadcast,** disabling the flooding of broadcast packets via `vlan2`, enter the following commands:

```
           awplus# configure terminal
   awplus(config)# interface vlan2
awplus(config-if)# no ip directed-broadcast
```

Related Commands ip forward-protocol udp
ip helper-address
show running-config

ip forward-protocol udp

This command enables you to control which UDP broadcasts will be forwarded to the helper address(es). A UDP broadcast will only be forwarded if the destination UDP port number in the packet matches one of the port numbers specified using this command.

Refer to the IANA site (www.iana.org) for a list of assigned UDP port numbers for protocols to forward using **ip forward-protocol udp**.

Use the **no** variant of this command to remove a port number from the list of destination port numbers that are used as the criterion for deciding if a given UDP broadcast should be forwarded to the IP helper address(es).

Syntax

```
ip forward-protocol udp <port>

no ip forward-protocol udp <port>
```

Parameter	Description
<port>	UDP Port Number.

Default The **ip forward-protocol udp** command is not enabled by default.

Mode Global Configuration

Usage Combined with the "ip helper-address" on page 25.27 in interface mode, the **ip forward-protocol udp** command in Global Configuration mode allows control of which protocols (destination port numbers) are forwarded. The **ip forward-protocol udp** command configures protocols for forwarding, and the **ip helper-address** command configures the destination address(es).

Note The types of UDP broadcast packets that the switch will forward are ONLY those specified by the **ip forward-protocol** command(s). There are not other UDP packet types that the IP helper process forwards by default.

Note The **ip forward-protocol udp** command does not support BOOTP / DHCP Relay. The **ip dhcp-relay** command must be used instead. For this reason, you may not configure UDP ports 67 and 68 with the **ip forward-protocol udp** command.

See "DHCP Relay Agent Introduction" on page 60.8 for information about DHCP Relay.

Examples To configure forwarding of packets on a UDP port, use the following commands:

```
        awplus# configure terminal
awplus(config)# ip forward-protocol udp <port>
```

To delete a UDP port from the UDP ports that the switch forwards, use the following commands:

```
        awplus# configure terminal
awplus(config)# no ip forward-protocol udp <port>
```

Validation Commands show running-config

Related Commands ip helper-address
ip directed-broadcast

ip gratuitous-arp-link

This command sets the Gratuitous ARP time limit for all switchports. The time limit restricts the sending of Gratuitous ARP packets to one Gratuitous ARP packet within the time in seconds.

Note This command specifies time between sequences of Gratuitous ARP packets, and time between individual Gratuitous ARP packets occurring in a sequence, to allow legacy support for older devices and interoperation between other devices that are not ready to receive and forward data until several seconds after linkup.

Additionally, jitter has been applied to the delay following linkup, so Gratuitous ARP packets applicable to a given port are spread over a period of 1 second so are not all sent at once. Remaining Gratuitous ARP packets in the sequence occur after a fixed delay from the first one.

Syntax

```
ip gratuitous-arp-link <0-300>

no ip gratuitous-arp-link
```

Parameter	Description
<0-300>	Specify the minimum time between sequences of Gratuitous ARPs and the fixed time between Gratuitous ARPs occurring in a sequence, in seconds. 0 disables the sending of Gratuitous ARP packets. The default is 5 seconds.

Default The default Gratuitous ARP time limit for all switchports is 5 seconds.

Mode Global Configuration

Usage Every switchport will send a sequence of 3 Gratuitous ARP packets to each VLAN that the switchport is a member of, whenever the switchport moves to the forwarding state. The first Gratuitous ARP packet is sent 1 second after the switchport becomes a forwarding switchport. The second and third Gratuitous ARP packets are each sent after the time period specified by the Gratuitous ARP time limit.

Additionally, the Gratuitous ARP time limit specifies the minimum time between the end of one Gratuitous ARP sequence and the start of another Gratuitous ARP sequence. When a link is flapping, the switchport's state is set to forwarding several times. The Gratuitous ARP time limit is imposed to prevent Gratuitous ARP packets from being sent undesirably often.

Examples To disable the sending of Gratuitous ARP packets, use the commands:

```
        awplus# configure terminal
awplus(config)# ip gratuitous-arp-link 0
```

To restrict the sending of Gratuitous ARP packets to one every 20 seconds, use the commands:

```
        awplus# configure terminal
awplus(config)# ip gratuitous-arp-link 20
```

Validation Commands show running-config

ip helper-address

This command adds a forwarding destination address for IP Helper to enable forwarding of User Datagram Protocol (UDP) broadcasts on an interface.

Use the **no** variant of this command to disable the forwarding od broadcast packets to specific addresses.

Syntax

```
ip helper-address <ip-addr>
no ip helper-address <ip-addr>
```

Parameter	**Description**
<ip-addr>	Forwarding destination IP address for IP Helper.

Default The destination address for the **ip helper-address** command is not configured by default.

Mode Interface Configuration for a VLAN interface.

Usage Combined with the ip forward-protocol udp command in global configuration mode, the **ip helper-address** command in interface mode allows control of which protocols (destination port numbers) are forwarded. The **ip forward-protocol udp** command configures protocols for forwarding, and the **ip helper-address** command configures the destination address(es).

The destination address can be a unicast address or a subnet broadcast address. The UDP destination port is configured separately with the **ip forward-protocol udp** command. If multiple destination addresses are registered then UDP packets are forwarded to each IP address added to an IP Helper. Up to 32 destination addresses may be added using IP Helper.

Note The types of UDP broadcast packets that the switch will forward are ONLY those specified by the **ip forward-protocol** command(s). There are no other UDP packet types that the IP helper process forwards by default.

Note The **ip helper-address** command does not support BOOTP / DHCP Relay. The **ip dhcp-relay** command must be used instead. For this reason, you may not configure UDP ports 67 and 68 with the **ip forward-protocol** command.

For information about DHCP Relay, see "DHCP Relay Agent Introduction" on page 60.8.

Examples The following example defines IPv4 address `192.168.1.100` as an IP Helper destination address to which to forward UDP broadcasts received on `vlan2`:

```
          awplus# configure terminal
  awplus(config)# interface vlan2
awplus(config-if)# ip helper-address 192.168.1.100
```

The following example removes IPv4 address 192.168.1.100 as an IP Helper destination address to which to forward UDP broadcasts received on vlan2:

```
        awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# no ip helper-address 192.168.1.100
```

Validation Commands show running-config

Related Commands ip forward-protocol udp
ip directed-broadcast

ip irdp

This command enables ICMP Router Discovery advertising on an interface. However, the interface does not send or process Router Discovery messages until at least one IP address is configured on the interface with the ip address command.

The **no** variant of this command disables ICMP Router Discovery advertisements on an IP interface. All transmitting and processing of Router Discovery messages ceases immediately on the interface.

Syntax

```
ip irdp
```

```
no ip irdp
```

Mode Interface Configuration for a VLAN interface.

Examples To enable Router Discovery advertisements on `vlan4`, use the following commands:

```
          awplus# configure terminal
  awplus(config)# interface vlan4
awplus(config-if)# ip irdp
```

To disable Router Discovery advertisements on `vlan4`, use the following commands:

```
          awplus# configure terminal
  awplus(config)# interface vlan4
awplus(config-if)# no ip irdp
```

Related Commands

ip address
show ip irdp
show ip irdp interface

ip irdp address preference

When multiple routers connected to a LAN are all sending Router Discovery advertisements, hosts need to be able to choose the best router to use. Therefore the IRDP defines a preference value to place in the Router Discovery advertisements. Hosts choose the router with the highest preference value.

This command sets the preference value to include in Router Discovery advertisements sent for the specified IP address.

The **no** variant of this command sets the preference for a specific address to the default of **0**.

Syntax

```
ip irdp address <ip-address> preference <0-2147483647>
```

```
no ip irdp address <ip-address> preference
```

Parameter	Description
<ip-address>	The IP address to be advertised with the specified preference value.
<0-2147483647>	The preference value advertised. A higher number increases the preference level for this address.

Default The default preference value is 0.

Mode Interface Configuration for a VLAN interface.

Examples To set the preference value to 3000 for the address 192.168.1.1 advertised on vlan5, use the following commands:

```
          awplus#  configure terminal
  awplus(config)#  interface vlan5
awplus(config-if)# ip irdp address 192.168.1.1 preference 3000
```

To set the preference value to the default of 0 for the address 192.168.1.1 advertised on vlan5, use the following commands:

```
          awplus#  configure terminal
  awplus(config)#  interface vlan5
awplus(config-if)# no ip irdp address 192.168.1.1 preference
```

Related Commands

ip irdp
ip irdp preference
show ip irdp interface

ip irdp broadcast

This command configures broadcast Router Discovery advertisements on an interface. The interface sends IRDP advertisements with the broadcast address (255.255.255.255) as the IP destination address.

The **no** variant of this command configures multicast Router Discovery advertisements on an interface. The interface sends IRDP advertisements with the all-system multicast address (224.0.0.1) as the IP destination address.

Syntax

```
ip irdp broadcast
```

```
no ip irdp broadcast
```

Mode Interface Configuration for a VLAN interface.

Examples To enable broadcast Router Discovery advertisements on `vlan13`, use the following commands:

```
          awplus#  configure terminal
   awplus(config)#  interface vlan13
awplus(config-if)#  ip irdp broadcast
```

To enable multicast Router Discovery advertisements on `vlan13`, use the following commands:

```
          awplus#  configure terminal
   awplus(config)#  interface vlan13
awplus(config-if)#  no ip irdp broadcast
```

Related Commands
ip irdp
ip irdp multicast
show ip irdp interface

ip irdp holdtime

This command sets the maximum length of time that the advertised addresses are to be considered as valid router addresses by hosts.

The **no** variant of this command resets the holdtime back to the default of 1800 seconds.

Syntax

```
ip irdp holdtime <0-9000>

no ip irdp holdtime
```

Parameter	Description
<0-9000>	The holdtime value in seconds of addresses advertised.

Default The IRDP holdtime is set to 1800 seconds (30 minutes) by default.

Mode Interface Configuration for a VLAN interface.

Examples To set the holdtime value of addresses advertised on `vlan2` to 4000 seconds, use the following commands:

```
          awplus# configure terminal
  awplus(config)# interface vlan2
awplus(config-if)# ip irdp holdtime 4000
```

To set the holdtime value of addresses advertised on `vlan2` back to the default, use the following commands:

```
          awplus# configure terminal
  awplus(config)# interface vlan2
awplus(config-if)# no ip irdp holdtime
```

Related Commands show ip irdp interface

ip irdp lifetime

This command sets the maximum length of time that hosts should consider the Router Discovery advertised addresses as valid router addresses. If you change the lifetime value, also change the **maxadvertisementinterval** and the **minadvertisementinterval** to maintain the following ratios:

```
lifetime=3 x maxadvertisementinterval
minadvertisementinterval=0.75 x maxadvertisementinterval
```

This command is synonymous with the **ip irdp hostname <*0-9000*>** command.

The **no** variant of this command sets the lifetime back to the default of 1800 seconds.

Syntax

```
ip irdp lifetime <0-9000>
no ip irdp lifetime
```

Parameter	Description
<0-9000>	Lifetime value in seconds of the advertised addresses.

Default The lifetime value is 1800 seconds.

Mode Interface Configuration for a VLAN interface.

Examples To set the lifetime value to 4000 seconds for addresses advertised on `vlan6`, use the following commands:

```
          awplus# configure terminal
  awplus(config)# interface vlan6
awplus(config-if)# ip irdp lifetime 4000
```

To set the lifetime value to the default of 1800 seconds for addresses advertised on `vlan6`, use the following commands:

```
          awplus# configure terminal
  awplus(config)# interface vlan6
awplus(config-if)# no ip irdp lifetime
```

Related Commands

ip irdp
ip irdp maxadvertinterval
ip irdp minadvertinterval
show ip irdp interface

ip irdp maxadvertinterval

This command sets the maximum time allowed between sending router advertisements from the interface. If you change the **maxadvertisementinterval** value, also change the **lifetime** and the **minadvertisementinterval** to maintain the following ratios:

```
lifetime=3 x maxadvertisementinterval
minadvertisementinterval=0.75 x maxadvertisementinterval
```

You cannot set the maximum advertisement interval below the minimum interval. If you are lowering the maximum interval to a value below the current minimum interval, you must change the minimum value first.

The **no** variant of this command sets the **maxadvertinterval** back to the default of 600 seconds.

Syntax

```
ip irdp maxadvertinterval <4-1800>
no ip irdp maxadvertinterval
```

Parameter	Description
<4-1800>	The maximum time, in seconds, between Router Discovery advertisements.

Default The IRDP maximum advertisement interval is set to 600 seconds (10 minutes) by default.

Mode Interface Configuration for a VLAN interface.

Examples To set the maximum interval between Router Discovery advertisements on vlan7 to 950 seconds, use the following commands:

```
           awplus# configure terminal
   awplus(config)# interface vlan7
awplus(config-if)# ip irdp maxadvertinterval 950
```

To set the maximum interval between advertisements on vlan7 back to the default, use the following commands:

```
           awplus# configure terminal
   awplus(config)# interface vlan7
awplus(config-if)# no ip irdp maxadvertinterval
```

Related Commands

ip irdp
ip irdp lifetime
ip irdp minadvertinterval
show ip irdp interface

ip irdp minadvertinterval

This command sets the minimum time allowed between sending router advertisements from the interface. If you change the **minadvertisementinterval** value, also change the **lifetime** and the **maxadvertisementinterval** to maintain the following ratios:

```
lifetime=3 x maxadvertisementinterval
minadvertisementinterval=0.75 x maxadvertisementinterval
```

You cannot set the minimum advertisement interval above the maximum interval. If you are raising the minimum interval to a value above the current maximum interval, you must change the maximum value first.

The **no** variant of this command sets the **minadvertinterval** back to the default of 450 seconds.

Syntax

```
ip irdp minadvertinterval <3-1800>
no ip irdp minadvertinterval
```

Parameter	Description
<3-1800>	The minimum time between advertisements in seconds.

Default The IRDP minimum advertisement interval is set to 450 seconds (7.5 minutes) by default.

Mode Interface Configuration for a VLAN interface

Examples To set the minimum interval between advertisements on `vlan4` to 900 seconds, use the following commands:

```
          awplus# configure terminal
  awplus(config)# interface vlan4
awplus(config-if)# ip irdp minadvertinterval 900
```

To set the minimum interval between advertisements on `vlan4` back to the default of 450 seconds, use the following commands:

```
          awplus# configure terminal
  awplus(config)# interface vlan4
awplus(config-if)# no ip irdp minadvertinterval
```

Related Commands

ip irdp
ip irdp lifetime
ip irdp maxadvertinterval
show ip irdp interface

ip irdp multicast

This command configures multicast Router Discovery advertisements on an interface. The interface sends IRDP advertisements with the all-system multicast address (224.0.0.1) as the IP destination address.

The **no** variant of this command configures broadcast Router Discovery advertisements on an interface. The interface sends IRDP advertisements with the broadcast address (255.255.255.255) as the IP destination address.

The multicast address is the default IP destination address for Router Discovery advertisements.

Syntax

```
ip irdp multicast

no ip irdp multicast
```

Mode Interface Configuration for a VLAN interface.

Examples To enable multicast Router Discovery advertisements on **vlan5**, use the following commands:

```
           awplus# configure terminal
   awplus(config)# interface vlan5
awplus(config-if)# ip irdp multicast
```

To enable broadcast Router Discovery advertisements on **vlan5**, use the following commands:

```
           awplus# configure terminal
   awplus(config)# interface vlan5
awplus(config-if)# no ip irdp multicast
```

Related Commands

ip irdp
ip irdp broadcast
show ip irdp interface

ip irdp preference

When multiple routers connected to a LAN are all sending Router Discovery advertisements, hosts need to be able to choose the best router to use. Therefore the IRDP defines a preference value to place in the Router Discovery advertisements. Hosts choose the router with the highest preference value.

This command sets the preference value to include in Router Discovery advertisements sent for the specified interface.

When this command is used, all IP addresses on the interface are assigned the same preference value, except the addresses that have specific preference value assignment using the command ip irdp address preference.

The **no** variant of this command sets the preference value to the default of 0.

Syntax

```
ip irdp preference <0-2147483647>

no ip irdp preference
```

Parameter	Description
<0-2147483647>	The preference value for the interface. A higher number increases the preference level for addresses on the specific interface.

Default The default preference value is 0.

Mode Interface Configuration for a VLAN interface.

Examples To set the preference of addresses advertised on `vlan6` to `500`, use the following commands:

```
awplus#           configure terminal
awplus(config)#   interface vlan6
awplus(config-if)# ip irdp preference 500
```

To set the preference value for addresses on `vlan6` back to the default of `0`, use the following commands:

```
awplus#           configure terminal
awplus(config)#   interface vlan6
awplus(config-if)# no ip irdp preference
```

Related Commands

ip irdp
ip irdp address preference
show ip irdp interface

ip local-proxy-arp

This command allows you to stop MAC address resolution between hosts within a private VLAN edge interface. Local Proxy ARP works by intercepting ARP requests between hosts within a subnet and responding with your device's own MAC address details instead of the destination host's details. This stops hosts from learning the MAC address of other hosts within its subnet through ARP requests.

Local Proxy ARP ensures that devices within a subnet cannot send traffic that bypasses Layer 3 routing on your device. This lets you monitor and filter traffic between hosts in the same subnet, and enables you to have control over which hosts may communicate with one another.

When Local Proxy ARP is operating on an interface, your device does not generate or forward any ICMP-Redirect messages on that interface. This command does not enable proxy ARP on the interface; see the ip proxy-arp command for more information on enabling proxy ARP.

The **no** variant of this command disables Local Proxy ARP to stop your device from intercepting and responding to ARP requests between hosts within a subnet. This allows the hosts to use MAC address resolution to communicate directly with one another. Local Proxy ARP is disabled by default.

Syntax

```
ip local-proxy-arp

no ip local-proxy-arp
```

Default Local proxy ARP is disabled by default

Mode Interface Configuration for a VLAN interface.

Examples To enable your device to apply Local Proxy ARP on the interface `vlan7`, use the following commands:

```
          awplus#  configure terminal
  awplus(config)#  interface vlan7
awplus(config-if)# ip local-proxy-arp
```

To disable your device to apply Local Proxy ARP on the interface `vlan7`, use the following commands:

```
          awplus#  configure terminal
  awplus(config)#  interface vlan7
awplus(config-if)# no ip local-proxy-arp
```

Related Commands

ip proxy-arp
show arp
show running-config

ip name-server

This command adds the IPv4 or the IPv6 address of a DNS server to the device's list of servers. The DNS client on your device sends DNS queries to devices on this list when trying to resolve a DNS hostname. Your device cannot resolve a hostname until you have added at least one server to this list. There is no limit on the number of servers you can add to the list.

The **no** variant of this command removes the DNS server from the list of servers.

Syntax

```
ip name-server <ip-addr>

no ip name-server <ip-addr>
```

Parameter	Description
<ip-addr>	The IP address to be advertised with the specified preference value, entered in the form `A.B.C.D` for an IPv4 address, or in the form `X:X::X.X` for an IPv6 address.

Mode Global Configuration

Usage When your device is using its DHCP client for an interface, it can receive Option 6 from the DHCP server. This option appends the name server list with more DNS servers. See Chapter 60, Dynamic Host Configuration Protocol (DHCP) Introduction for more information about DHCP and DHCP options.

See "DNS Relay" on page 24.10 for more information about DNS Relay to map IPv4 and IPv6 addresses to name servers to maintain a a database of hostname-to-address mappings. Also see "DNS Client" on page 24.9 for information about DNS Client configuration commands.

Example To allow your device to send DNS queries to a DNS server with the IPv4 address `10.10.10.5`, use the commands:

```
        awplus# configure terminal
awplus(config)# ip name-server 10.10.10.5
```

To allow your device to send DNS queries to a DNS server with the IPv6 address `2001:0db8:010d::1`, use the commands:

```
        awplus# configure terminal
awplus(config)# ip name-server 2001:0db8:010d::1
```

Related Commands ip domain-list
ip domain-lookup
ip domain-name
show ip name-server

ip proxy-arp

This command enables Proxy ARP responses to ARP requests on an interface. When enabled, your device intercepts ARP broadcast packets and substitutes its own physical address for that of the remote host. By responding to the ARP request, your device ensures that subsequent packets from the local host are directed to its physical address, and it can then forward these to the remote host.

Your device responds only when it has a specific route to the address being requested, excluding the interface route that the ARP request arrived from. It ignores all other ARP requests. See the ip local-proxy-arp command about enabling your device to respond to other ARP messages.

The **no** variant of this command disables Proxy ARP responses on an interface. Proxy ARP is disabled by default.

Syntax

```
ip proxy-arp

no ip proxy-arp
```

Default Proxy ARP is disabled by default.

Mode Interface mode for a VLAN interface.

Examples To enable your device to Proxy ARP on the interface `vlan13`, use the following commands:

```
          awplus# configure terminal
  awplus(config)# interface vlan13
awplus(config-if)# ip proxy-arp
```

To disable your device to Proxy ARP on the interface `vlan13`, use the following commands:

```
          awplus# configure terminal
  awplus(config)# interface vlan13
awplus(config-if)# no ip proxy-arp
```

Related Commands

arp (IP address MAC address)
ip local-proxy-arp
show arp
show running-config

ip redirects

This command enables ICMP redirects for a device.

Use the **no** variant of this command to disable the sending of ICMP redirects for a device.

Syntax

```
ip redirects

no ip redirects
```

Default ICMP redirects are disabled by default.

Mode Global Configuration

Usage ICMP redirect messages are used to notify hosts that a better route is available to a destination. ICMP redirects are used when a packet is routed into the switch on the same interface that the packet is routed out of the switch. ICMP redirects are also used when the subnet or network of the source address is on the same subnet or network as the next-hop address for a packet.

Use the **ip redirects** command to allow the sending of ICMP redirects whenever the switch receives a packet that is routed on the same interface that the packet was sent on.

Use the **no** variant of this command to disallow the sending of ICMP redirects whenever the switch receives a packet that is routed on the same interface that the packet was sent on.

Examples To enable ICMP redirects, use the following commands:

```
        awplus# configure terminal
awplus(config)# ip redirects
```

To disable ICMP redirects, use the following commands:

```
        awplus# configure terminal
awplus(config)# no ip redirects
```

optimistic-nd

Use this command to enable the optimistic neighbor discovery feature for both IPv4 and IPv6.

Use the **no** variant of this command to disable the optimistic neighbor discovery feature.

Syntax

```
optimistic-nd

no optimistic-nd
```

Default The optimistic neighbor discovery feature is enabled by default.

Mode Interface Configuration for a VLAN interface.

Usage The optimistic neighbor discovery feature allows the switch, after learning an IPv4 or IPv6 neighbor, to refresh the neighbor before the neighbor is deleted from the hardware L3 switching table. The neighbor is put into the 'stale' state in the software switching table if is it not refreshed, then the 'stale' neighbors are deleted from the hardware L3 switching table.

The optimistic neighbor discovery feature enables the switch to sustain L3 traffic switching to a neighbor without interruption. Without the optimistic neighbor discovery feature enabled L3 traffic is interrupted when a neighbor is 'stale' and is then deleted from the L3 switching table.

If a neighbor receiving optimistic neighbor solicitations does not answer optimistic neighbor solicitations with neighbor advertisements, then the neighbor will be put into the 'stale' state, and subsequently deleted from both the software and the hardware L3 switching tables.

Examples To enable the optimistic neighbor discovery feature on `vlan100`, use the following commands:

```
          awplus# configure terminal
  awplus(config)# interface vlan100
awplus(config-if)# optimistic-nd
```

To disable the optimistic neighbor discovery feature on `vlan100`, use the following commands:

```
          awplus# configure terminal
  awplus(config)# interface vlan100
awplus(config-if)# no optimistic-nd
```

Validation Commands show running-config

ping

This command sends a query to another IPv4 host (send Echo Request messages).

Syntax x600

```
ping [ip] <host> [broadcast] [df-bit {yes|no}] [interval <0-128>]
   [pattern <hex-data-pattern>] [repeat {<1-2147483647>|continuous}]
   [size <36-18024>] [source <ip-addr>] [timeout <1-65535>] [tos <0-
   255>]
```

Parameter	Description
`<host>`	The destination IP address or hostname.
`broadcast`	Allow pinging of a broadcast address.
`df-bit`	Enable or disable the do-not-fragment bit in the IP header.
`interval <0-128>`	Specify the time interval in seconds between sending ping packets. The default is 1.
`pattern <hex-data-pattern>`	Specify the hex data pattern.
`repeat`	Specify the number of ping packets to send.
`<1-2147483647>`	Specify repeat count. The default is 5.
`continuous`	Continuous ping
`size <36-18024>`	The number of data bytes to send, excluding the 8 byte ICMP header. The default is 56 (64 ICMP data bytes).
`source <ip-addr>`	The IP address of a configured IP interface to use as the source in the IP header of the ping packet.
`timeout <1-65535>`	The time in seconds to wait for echo replies if the ARP entry is present, before reporting that no reply was received. If no ARP entry is present, it does not wait.
`tos <0-255>`	The value of the type of service in the IP header.

Mode User Exec and Privileged Exec

Example To ping the IP address `10.10.0.5` use the following command:

```
awplus# ping 10.10.0.5
```

router ip irdp

This command globally enables ICMP Router Discovery (IRDP) advertisements on your device. However, your device does not send or process IRDP messages until at least one interface is configured to use IP and has had IRDP enabled on the interface with the ip irdp command.

The **no** variant of this command globally disables IRDP advertisements on the device. All interfaces immediately stop transmitting and processing Router Discovery messages.

Syntax

```
router ip irdp

no router ip irdp
```

Mode Global Configuration

Examples To enable Router Discovery advertisements on your device, use the following commands:

```
        awplus# configure terminal
awplus(config)# router ip irdp
```

To disable Router Discovery advertisements on your device, use the following commands:

```
        awplus# configure terminal
awplus(config)# no router ip irdp
```

Related Commands ip irdp
show ip irdp

show arp

Use this command to display entries in the ARP routing and forwarding table—the ARP cache contains mappings of IP addresses to physical addresses for hosts. To have a dynamic entry in the ARP cache, a host must have used the ARP protocol to access another host.

For information on output options, see “Controlling “show” Command Output” on page 1.41.

User Exec and Privileged Exec

Mode

Usage Running this command with no additional parameters, will display all entries in the ARP routing and forwarding table.

Example To display all ARP entries in the ARP cache, use the following command:

```
awplus# show arp
```

Output Figure 25-3: Example output from the **show arp** command

```
awplus# show arp

 IP Address       MAC Address    Interface    Port        Type
192.168.10.2     0015.77ad.fad8  vlan1       port1.0.1   dynamic
192.168.20.2     0015.77ad.fa48  vlan2       port1.0.2   dynamic
192.168.1.100    00d0.6b04.2a42  vlan2       port1.0.8   static
```

Table 25-2: Parameters in the output of the **show arp** command

Parameter	Meaning
`IP Address`	IP address of the network device this entry maps to.
`MAC Address`	Hardware address of the network device.
`Interface`	Interface over which the network device is accessed.
`Port`	Physical port that the network device is attached to.
`Type`	Whether the entry is a static or dynamic entry. Static entries are added using the arp (IP address MAC address) command. Dynamic entries are learned from ARP request/reply message exchanges.

Related Commands arp (IP address MAC address)
clear arp-cache

show debugging ip dns forwarding

Use this command to display the DNS Relay debugging status. DNS Relay debugging is set using the **debug ip dns forwarding** command.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax

```
show debugging ip dns forwarding
```

Mode User Exec and Privileged Exec

Example To display the DNS Relay debugging status, use the command:

```
awplus# show debugging ip dns forwarding
```

Output Figure 25-4: Example output from the **show debugging ip dns forwarding** command

```
DNS Relay debugging status:
  debugging is on
```

Related Commands debug ip dns forwarding

show debugging ip packet

Use this command to show the IP interface debugging status. IP interface debugging is set using the **debug ip packet interface** command.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax

```
show debugging ip packet
```

Mode

User Exec and Privileged Exec

Example

To display the IP interface debugging status when the terminal monitor off, use the command:

```
awplus# terminal no monitor
awplus# show debug ip packet
```

Output

Figure 25-5: Example output from the **show debugging ip packet** command with **terminal monitor** off

```
IP debugging status:
interface all tcp (stopped)
interface vlan1 arp verbose (stopped)
```

Example

To display the IP interface debugging status when the terminal monitor is on, use the command:

```
awplus# terminal monitor
awplus# show debug ip packet
```

Output

Figure 25-6: Example output from the **show debugging ip packet** command with **terminal monitor** on

```
IP debugging status:
interface all tcp (running)
interface vlan1 arp verbose (running)
```

Related Commands

debug ip packet interface
terminal monitor

show hosts

This command shows the default domain, domain list, and name servers configured on your device.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax show hosts

Mode User Exec and Privileged Exec

Example To display the default domain, use the command:

```
awplus# show hosts
```

Output Figure 25-7: Example output from the **show hosts** command

```
Default domain is mycompany.com
Domain list: company.com
Name/address lookup uses domain service
Name servers are 10.10.0.2 10.10.0.88
```

Related Commands ip domain-list
ip domain-lookup
ip domain-name
ip name-server

show ip dns forwarding

Use this command to display the DNS Relay status.

Syntax

```
show ip dns forwarding
```

Mode User Exec and Privileged Exec

Examples To display the DNS Relay status, use the command:

```
awplus# show ip dns forwarding
```

Output Figure 25-8: Example output from the **show ip dns forwarding** command

```
Servers           Forwards      Fails
192.168.1.1             12          0
192.168.1.2              6          3
```

Related Commands ip dns forwarding

show ip dns forwarding cache

Use this command to display the DNS Relay name resolver cache.

Syntax `show ip dns forwarding cache`

Mode User Exec and Privileged Exec

Examples To display the DNS Relay name resolver cache, use the command:

```
awplus# show ip dns forwarding cache
```

Output Figure 25-9: Example output from the **show ip dns forwarding cache** command

```
Host                                     Address                 Expires Flags
www.example.com                          192.168.1.1                 180
mail.example.com                         www.example.com             180 CNAME
www.example.com                          192.168.1.1                 180 REVERSE
mail.example.com                         192.168.1.5                 180
```

Related Commands ip dns forwarding cache

show ip domain-list

This command shows the domains configured in the domain list. The DNS client uses the domains in this list to append incomplete hostnames when sending a DNS enquiry to a DNS server.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax

```
show ip domain-list
```

Mode User Exec and Privileged Exec

Example To display the list of domains in the domain list, use the command:

```
awplus# show ip domain-list
```

Output Figure 25-10: Example output from the **show ip domain-list** command

```
alliedtelesis.com
mycompany.com
```

Related Commands ip domain-list
ip domain-lookup

show ip domain-name

This command shows the default domain configured on your device. When there are no entries in the DNS list, the DNS client appends this domain to incomplete hostnames when sending a DNS enquiry to a DNS server.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax

```
show ip domain-name
```

Mode User Exec and Privileged Exec

Example To display the default domain configured on your device, use the command:

```
awplus# show ip domain-name
```

Output Figure 25-11: Example output from the **show ip domain-name** command

```
alliedtelesis.com
```

Related Commands ip domain-name
ip domain-lookup

show ip forwarding

Use this command to display the IP forwarding status.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax `show ip forwarding`

Mode User Exec and Privileged Exec

Example

```
awplus# show ip forwarding
```

Output Figure 25-12: Example output from the **show ip forwarding** command

```
awplus#show ip forwarding
IP forwarding is on
```

show ip interface

Use this command to display information about interfaces and the IP addresses assigned to them. To display information about a specific interface, specify the interface name with the command.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax `show ip interface [<interface-list>] [brief]`

Parameter	Description
<interface-list>	The interfaces to display information about. An interface-list can be: ■ an interface, e.g. `vlan2` ■ a continuous range of interfaces separated by a hyphen, e.g. `vlan2-8` or `vlan2-vlan5` ■ a comma-separated list of interfaces or interface ranges, e.g. `vlan2,vlan5,vlan8-10` The specified interfaces must exist.

Mode User Exec and Privileged Exec

Examples To show the IP addresses assigned to `vlan2` and `vlan3`, use the command:

```
awplus# show ip interface vlan2-3 brief
```

Output Figure 25-13: Example output from the **show ip interface brief** command

```
Interface               IP-Address        Status            Protocol
port1.0.2               unassigned        admin up          down
vlan1                   192.168.1.1       admin up          running
vlan2                   192.168.2.1       admin up          running
vlan3                   192.168.3.1       admin up          running
vlan8                   unassigned        admin up          down
```

Figure 25-14: "Controlling "show" Command Output" on page 1.41

show ip irdp

This command displays whether IRDP is globally enabled on your device, and the status of the debugging modes.

If the **debug ip irdp** command has been set with the **detail** parameter then the **both** parameter is also set and the output will show "packet debugging mode is all".

For information on output options, see “Controlling “show” Command Output” on page 1.41.

Syntax

```
show ip irdp
```

Mode User Exec and Privileged Exec

Example To display global IRDP configuration, use the command:

```
awplus# show ip irdp
```

Output

Figure 25-15: Example output from the **show ip irdp** command

```
IRDP is enabled
  event debugging is disabled
  nsm debugging is disabled
  packet debugging mode is disabled
```

Figure 25-16: Example output from the **show ip irdp** command with **debug ip irdp detail** set

```
IRDP is enabled
  event debugging is disabled
  nsm debugging is disabled
  packet debugging mode is all
```

Figure 25-17: Example output from the **show ip irdp** command with **debug ip irdp both** set

```
IRDP is enabled
  event debugging is disabled
  nsm debugging is disabled
  packet debugging mode is both
```

Related Commands

debug ip irdp
router ip irdp

show ip irdp interface

This command displays the configuration of IRDP on all interfaces, or for a specified interface.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax `show ip irdp interface [<interface-name>]`

Parameter	Description
<interface-name>	Displays the interface status and configuration details of the specified interface.

Mode User Exec and Privileged Exec

Example To display the IRDP configuration for `vlan4`, use the command:

```
awplus# show ip irdp interface vlan4
```

Output Figure 25-18: Example output from the **show ip irdp interface** command

```
vlan13 is up, line protocol is up
ICMP Router Discovery Protocol
  Sending mode          multicast
  Router Lifetime       1350 seconds
  Default Preference    0
  Min Adv Interval      450 seconds
  Max Adv Interval      600 seconds
  Next advertisement in 551 seconds
  Non default prefix preferences
     192.168.1.1        preference     25000

  In packets                  0                 Out packets                3
  In bad packets              0                 Out bad packets            0
  In good packets             0                 Out good packets           3
  In ignored packets          0
```

Table 25-3: Parameters in the output of the **show ip irdp interface** command

Parameter	Description
`Sending mode`	Whether this interface is sending broadcast or multicast router advertisements. This means the destination IP address of router advertisements will be either the multicast address 224.0.0.1, or the broadcast address 255.255.255.255.
`Router Lifetime`	The lifetime value set for router advertisements sent from this interface. This is the maximum time that other devices should treat the advertised address as valid.
`Default Preference`	The preference value for IP addresses as default router addresses, relative to other router addresses on the same subnet. This preference value is used for all IP addresses on this interface, except for those listed under the heading "non default prefix preferences".
`Min Adv Interval`	Minimum time allowed between sending router advertisements from this interface.

Table 25-3: Parameters in the output of the **show ip irdp interface** command

Parameter	Description
`Max Adv Interval`	Maximum time allowed between sending router advertisements from this interface.
`Non default prefix preferences`	List of the IP addresses on this interface that have been set with a specific router preference value. These addresses use the preference value listed beside them, rather than the interface's default preference value.
`In packets`	The total number of packets received by IRDP on this interface. IRDP processes all ICMP packets received on this interface.
`Out packets`	The number of packets sent by IRDP on this interface.
`In bad packets`	The number of packets received by IRDP that it has discarded because they do not conform or corrupted.
`Out bad packets`	The number of packets that IRDP generated but failed to send to the network layer.
`In good packets`	The number of packets received and processed by IRDP.
`Out good packets`	The number of packets generated and successfully sent by IRDP.
`In ignored packets`	The number of incoming packets ignored, like ICMP packets other than IRDP.

Related Commands

ip irdp
show ip irdp

show ip name-server

This command displays the list of DNS servers your device sends DNS requests to with assigned IPv4 and IPv6 addresses. This is a static list configured using the ip name-server command.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax

```
show ip name-server
```

Mode User Exec and Privileged Exec

Example To display the list of DNS servers that your device sends DNS requests to, use the command:

```
awplus# show ip name-server
```

Output Figure 25-19: Example output from the **show ip name-server** command

```
Nameservers:
 10.10.0.123
 10.10.0.124
 2001:0db8:010d::1
 2001:0db8:010d::2
```

Related Commands ip domain-lookup
ip name-server

tcpdump

Use this command to start a tcpdump, which gives the same output as the Unix-like **tcpdump** command to display TCP/IP traffic. Press `<ctrl> + c` to stop a running tcpdump.

Parameter	Description
<line>	Specify the dump options. For more information on the options for this placeholder see URL http://www.tcpdump.org/tcpdump_man.html

Mode Privileged Exec

Example To start a tcpdump running to capture IP packets, enter the command:

```
awplus# tcpdump ip
```

Output Figure 25-20: Example output from the **tcpdump** command

```
03:40:33.221337 IP 192.168.1.1 > 224.0.0.13: Hello, length: 34
1 packets captured
2 packets received by filter
0 packets dropped by kernel
```

Related Commands debug ip packet interface

traceroute

Use this command to trace the route to the specified IPv4 host.

Parameter	Description
<ip-addr>	The destination IPv4 address. The IPv4 address uses the format A.B.C.D.
<hostname>	The destination hostname.

Mode User Exec and Privileged Exec

Example

```
awplus# traceroute 10.10.0.5
```

undebug ip packet interface

This command applies the functionality of the no debug ip packet interface command on page 25.11.

undebug ip irdp

This command applies the functionality of the no debug ip irdp command on page 25.13.

Chapter 26: IPv6 Introduction

Introduction

This chapter describes the main features of IPv6, the switch's implementation of IPv6 and how to configure and operate IPv6 on the switch.

This chapter describes the following IPv6 features:

- linking together networks that run IPv6.
- allowing address autoconfiguration of hosts connected to the switch.

Overview

IPv6 is the next generation of the Internet Protocol (IP). It has primarily been developed to solve the problem of the eventual exhaustion of the IPv4 address space, but also offers other enhancements. IPv6 addresses are 16 bytes long, in contrast to IPv4's 4 byte addresses. Other features of IPv6 include:

- Address structure improvements:
    - globally unique addresses with more levels of addressing hierarchy to reduce the size of routing tables
    - autoconfiguration of addresses by hosts
    - improved scalability of multicast routing by adding a "scope" field to multicast addresses
    - a new type of address, the "anycast address", which sends packets to any one of a group of devices
- Removes the need for packet fragmentation en-route, by dynamic determination of the largest packet size that is supported by every link in the path. A link's MTU (Maximum Transmission Unit) must be at least 1280 bytes, compared with 576 bytes for IPv4.
- Includes a Traffic Class that allow packets to be labelled with an appropriate priority. If the network becomes congested, the lowest priority packets are dropped.
- Includes Flow labels that indicate to intermediate switches and routers that packets are part of a flow, and that a particular flow requires a particular type of service. This feature enables, for example, real-time processing of data streams. It also increases routing speed because the forwarding router need only check the flow label, not the rest of the header. The handling indicated by the flow label can be done by the IPv6 Hop-by-Hop header, or by a separate protocol such as RSVP.
- Mandatory authentication and data integrity protocols through IPsec. IPsec is optional in IPv4.

IPv6 Addresses and Prefixes

IPv6 addresses are hexadecimal, and are made up of eight pairs of octets separated by colons. An example of a valid address is **2001:0db8:0000:0000:0260:0000:97ff:64aa**. In the interests of brevity, addresses can be abbreviated in two ways:

- Leading zeros can be omitted, so this address can be written as **2001:db8:0:0:260:0:97ff:64aa**.
- Consecutive zeros can be replaced with a double colon, so this address can be written as **2001:db8::260:0:97ff:64a**. Note that a double colon can replace any number of consecutive zeros, but an address can contain only one double colon.

Like IPv4 addresses, a proportion of the left most bits of the IPv6 address can be used to indicate the subnet, rather than a single node. This part of the address is called the *prefix*. Prefixes provide the equivalent functionality to a subnet mask in IPv4, allowing a subnet to be addressed, rather than a single node. If a prefix is specified, the IPv6 address is followed by a slash and the number of bits that represent the prefix. For example, **2001::/16** indicates that the first 16 bits (**2001**) of the address **2001:0:0:0:0:0:0:0** represent the prefix.

Like IPv4 addresses, IPv6 addresses are attached to interfaces.

Note RFC 3849 allocates the prefix **2001:0db8::/32** for documentation purposes.

Address types

IPv6 supports the following address types:

- Unicast
- Multicast
- Anycast

Unicast addresses

A unicast address is attached to a single interface and delivers packets only to that interface. The following special addresses have been defined:

- IPv4-compatible and IPv4-mapped addresses. IPv4-compatible addresses are used to tunnel IPv6 packets across an IPv4 network. IPv4-mapped addresses are used by an IPv6 host to communicate with an IPv4 host. The IPv6 host addresses the packet to the mapped address.
- Link-local addresses can be used on the local network on which the interface is attached. The link-local prefix is **fe80::/10**. Different interfaces on a device may have the same link-local address. The switch will automatically generate a link-local address for all interfaces that are using IPv6. Commands entered to configure link-local addresses that match any automatically generated link-local addresses by the switch will not be executed. Enter the **show ipv6 interface** command to display automatically generated link-local addresses not shown in the running-config. Automatically generated link-local addresses contain the last six hexadecimal numbers of the MAC address for a given interface.

- The Loopback address, consisting of **::1**, which is the equivalent of the IPv4 loopback address, and allows a host to send packets to itself.
- The Unspecified address, consisting of **::**, which is the equivalent of the IPv4 unspecified address, and is used as a source address by hosts during the autoconfiguration process.

Anycast addresses

An *anycast* address is a unicast address that is attached to more than one interface. If a packet is sent to an anycast address it is delivered to the nearest interface with that address, with the definition of "nearest" depending on the protocol used for routing.

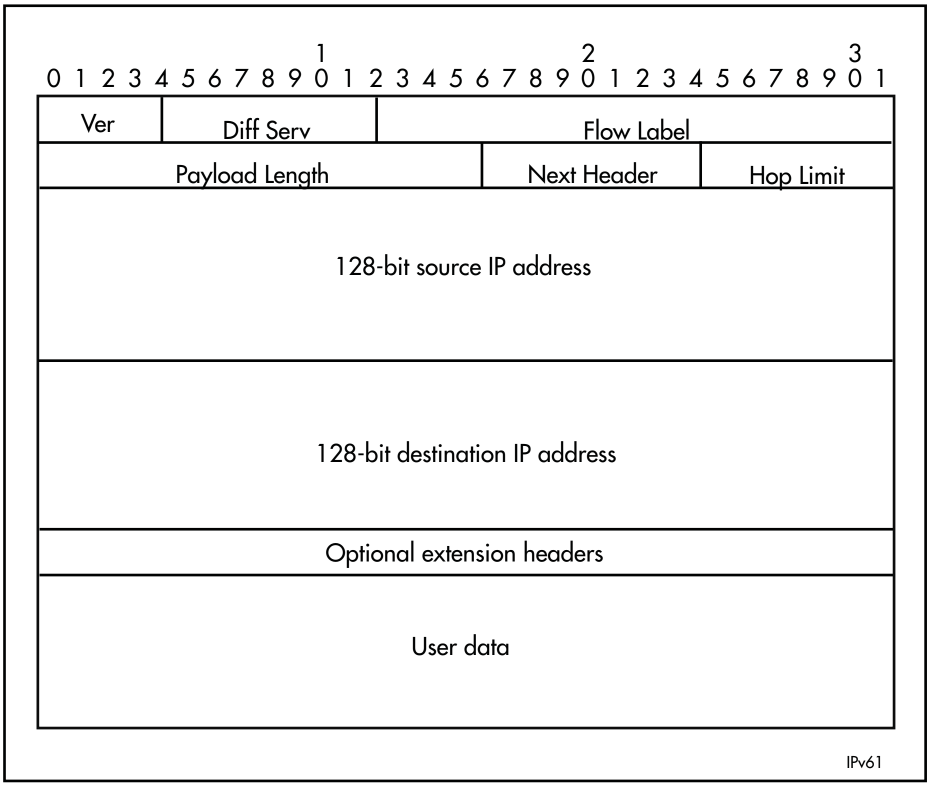
Anycast addresses can be assigned to routers only, and packets cannot originate from an anycast address. A router must be configured to know that it is using an anycast address because the address format cannot be distinguished from that of a unicast address.

Only one anycast address has been predefined: the subnet-router address. The subnet-router address sends messages to the nearest router on a subnet and consists of the subnet's prefix followed by zeros.

IPv6 Headers

The basic unit of data sent through an internet is called a *packet* in IPv6. A packet consists of a *header* followed by the *data*. The following figure shows the IPv6 packet.

Figure 26-1: IPv6 packet



.

Table 26-1: IPv6 packet - Field Description

Field	Function
Ver	Version of the IP protocol that created the packet. For IPv6, this field has a value of 6.
Differentiated Services	8-bit value that contains the 6-bit DSCP and is used to prioritize traffic as part of a Quality of Service system. For more information, see "Differentiated Services Architecture" on page 35.4. Additional information can be found in RFC 2474, *Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers*.
Flow Label	20-bit value that indicates the data flow to which this packet belongs.This flow may be handled in a particular way.
Payload Length	Length of the user data portion of the packet. If the data payload is larger than 64 kB, the length is given in the optional "Jumbo Payload" header and the Payload Length header is given a value of zero.
Next Header	Number that indicates the type of header that immediately follows the basic IP header. This header type may be an optional IPv6 extension header, a relevant IPv4 option header, or another protocol, such as TCP or ICMPv6. The IPv6 extension header values are: **0** (Hop-by-Hop Options Header) **43** (IPv6 Routing Header) **44** (IPv6 Fragment Header) **50** (Encapsulating Security Payload) **51** (IPv6 Authentication Header) **59** (No Next Header) **60** (Destination Options Header)
Hop Limit	Field that is the equivalent of the IPv4 Time To Live field, measured in hops.
Source IP address	128-bit IPv6 address of the sender.
Destination IP address	128-bit IPv6 address of the recipient.
Optional extension headers	Headers for less-frequently used information.
User data	Payload.

Basic IPv6 header structure

The headers contain information necessary to move the packet across the internet. They must be able to cope with missing and duplicated packets as well as possible fragmentation (and reassembly) of the original packet.

IPv6 headers are twice as long as IPv4 headers (40 bytes instead of 20 bytes) and contain four times the address space size (128 bits instead of 32 bits).

They no longer contains the header length, identification, flags, fragment offset, and header checksum fields. Some of these options are placed in extension headers. The Time To Live field is replaced with a hop limit, and the IPv4 Type of Service field is replaced with a Differentiated Services field. The Differentiated Services field contains the DSCP bits, used in a Quality of Service (QoS) regime. The following table explains IPv4 header fields that changed in IPv6.

Changed Field	Description
Type of Service	The type of service that a connection should receive is indicated in IPv6 by the Flow Label field in the IPv6 header.
Fragmentation information (the Identification field, the Flags field and the Fragment Offset field)	In most cases fragmentation does not occur in IPv6. If it does, packets are fragmented at their source and not en route. Therefore, the fragmentation information is contained in an extension header to reduce the size of the basic IPv6 header.
Header Checksum	This option has not been provided in IPv6. This is because transport protocols implement checksums and because of the availability of the IPsec authentication header (AH) in IPv6.
Options	Extension headers handle all the optional values associated with IPv6 packets. The biggest advantage of this scheme is that the size of the basic IP header is a constant.

Extension headers

IPv6 implements many of the less commonly used fields in the IPv4 header (or their equivalents) as extension headers, which are placed after the basic IPv6 header. The length of each header must be a multiple of 8 bytes.

The first extension header is identified by the Next Header field in the basic IPv6 header. Any subsequent extension headers are identified by an 8-bit "Next Header" value at the beginning of the preceding extension header.

IPv6 nodes that originate packets are required to place extension headers in a specific order:

1. The basic IPv6 header. This must come immediately before the extension headers.
2. The Hop-by-Hop header. This specifies options that must be examined by every node in the routing path.
3. A Destination Options header. This is used to specify options to be processed by the first destination or final destination. The destination options header is the only extension header that may be present more than once in the IPv6 packet.
4. The Routing header. This enables a static path to be specified for the packet, if the dynamically-determined path is undesirable.
5. The Fragment header. This indicates that the source node has fragmented the packet, and contains information about the fragmentation.
6. The Authentication header (AH). This verifies the integrity of the packet and its headers.
7. The Encapsulating Security Payload (ESP) header. This encrypts a packet and verifies the integrity of its contents.

8. The Upper Layer Protocol header. This indicates which protocol a higher layer (such as the transport layer) is to process the packet with (for example, TCP).

The Internet Control Message Protocol (ICMPv6)

The Internet Control Message Protocol, ICMPv6, provides a mechanism for error reporting and route discovery and diagnostics. It also conveys information about multicast group membership, a function that is carried out by the Internet Group Management Protocol (IGMP) in IPv4, and performs address resolution, which the Address Resolution Protocol (ARP) performs in IPv4.

Significant aspects of ICMPv6 include neighbor discovery, which enables one device in a network to find out about other nearby devices; and stateless address autoconfiguration, which allows a device to dynamically determine its own IPv6 address.

ICMPv6 is also used to support the Ping v6 (*Packet Internet Groper*) and Trace route v6 functions that are used to verify the connections between networks and network devices. Ping is used to test the connectivity between two network devices to determine whether each network device can "see" the other device. Trace route is used to discover the route used to pass packets between two systems running the IP protocol.

Both of these functions operate almost identically in IPv4 and IPv6. For more information, see "Ping" on page 24.17.

Neighbor Discovery

Neighbor discovery is an ICMPv6 function that enables a router or a host to identify other devices on its links. This information is then used in address autoconfiguration, to redirect a node to use a more appropriate router if necessary, and to maintain reachability information with its neighbors.

The IPv6 Neighbor Discovery protocol is similar to a combination of the IPv4 protocols ARP, ICMP Router Discovery and ICMP Redirect.

The following table describes packet types involved with neighbor discovery.

Packet Type	Description
router solicitation	Packet in which a host sends out a request for routers to generate advertisements.
router advertisement	Allows routers to advertise their presence and other network parameters. A router sends an advertisement packet in response to a solicitation packet from a host.
neighbor solicitation	Packet in which a node sends a packet to determine the link layer address of a neighbor or to verify that a neighbor is still active.
neighbor advertisement	A response to a neighbor solicitation packet. These packets are also used to notify neighbors of link layer address changes.
redirect	Informs hosts of a better first hop.

To comply with Section 6.2.1 of RFC 2461, *IPv6 Neighbor Discovery*, the router does not generate router advertisements by default. See "Neighbor Discovery" on page 26.7 for instructions about enabling advertisements.

The following table explains packet types and services.

Packet Type	Description
address resolution	A method for carrying out address autoconfiguration, and is achieved using the Neighbor Solicitation Message and the Neighbor Advertisement Message.
router and prefix discovery	On connection to a link, a node needs to know the address of a router that the node can use to reach the rest of the world. The node also needs to know the prefix (or prefixes) that define the range of IP addresses on its link that it can reach without going through a router. Routers use ICMP to convey this information to hosts, by means of router advertisements. The message may have an option attached (the *source link address* option), which enables the receiving node to respond directly to the router, without performing a neighbor solicitation.
immediate information	The configuration of a router includes a defined frequency at which unsolicited advertisements are sent. If a node wants to obtain information about the nearest router immediately, rather than waiting for the next unsolicited advertisement, the node can send a router solicitation message. Each router that receives the solicitation message sends a router advertisement specifically to the node that sent the solicitation.
redirection	If a node is aware of more than one router that it can use to connect to wider networks, the router to which it sends packets by default does not always represent the most desirable route. ICMPv6 uses the redirect packet to communicate a more effective path to the node.
Neighbor Unreachability Detection (NUD)	A node may issue solicitation requests to determine whether a path is still viable, or may listen in on acknowledgement packets of higher layer protocols, such as TCP. If the node determines that a path is no longer viable, it attempts to establish a new link to the neighbor, or to re-establish the previous link. NUD can be used between any two devices in the network, independent of whether the devices are acting as hosts or routers.

Stateless address autoconfiguration

Stateless address autoconfiguration allows an IPv6-aware device to be plugged into a network without manual configuration with an IP address. This plug and play functionality results in networks that are easier to set up and modify, and simplifies the process of shifting to use a new Internet Service Provider (ISP).

Stateless address autoconfiguration is achieved in a series of steps. Routers and hosts perform the first three steps, which autoconfigure a link-local address. A global address is autoconfigured in the last three steps, which only hosts perform.

On the router or host

1. During system start-up, the node begins autoconfiguration by generating a link-local address for the interface. A link-local address is formed by adding the interface ID to the link-local prefix **fe80::/10** (reference RFC 3513).

Note Different interfaces on a device may have the same link-local address. The switch will automatically generate a link-local address for all interfaces that are using IPv6. Commands entered to configure link-local addresses that match any automatically generated link-local addresses by the switch will not be executed. Enter the show ipv6 interface command to display automatically generated link-local addresses not shown in the running-config. Automatically generated link-local addresses contain the last six hexadecimal numbers of the MAC address for a given interface.

2. The node then transmits a neighbor solicitation message to this address. If the address is already in use, the node that the address belongs to replies with a neighbor advertisement message. The autoconfiguration process stops and manual configuration of the node is then required.

3. If no neighbor advertisement is received, the node concludes that the address is available and assigns it to the chosen interface.

On the host

1. The node then sends one or more router solicitations to detect if any routers are present. Any routers present responds with a router advertisement.

   If no router advertisement is received, the node tries to use DHCP to obtain an address and other configuration information. If no DHCP server responds, the node continues using the link-level address

   If a router advertisement is received, this message informs the node how to proceed with the auto configuration process. The prefix from the router advertisement, if received, is added to the link-level address to form the global unicast IP address.

2. This address is then assigned to the network interface.

   If routers are present, the node continues to receive router advertisements. The node updates its configuration when there are changes in the router advertisements.

Integration of IPv4 and IPv6

IPv6 has been designed in such a way that a smooth transition from IPv4 is possible. The most effective way to ensure this is to use a *dual IP stack*. A node configured as a dual stack system has both a 128-bit IPv6 address and a 32-bit IPv4 address, and so can communicate with nodes running IPv4 and those running IPv6.

Another aspect of the transition is to *tunnel* IPv6 packets through an IPv4 network. IPv6 packets are tunnelled simply by encapsulating the IPv6 packet within an IPv4 datagram, and identifying that this datagram is an encapsulated IPv6 packet by giving the datagram a protocol value of 41.

IPv6 on your Switch

This section describes the switch's support for IPv6, and how to configure IPv6 on the switch.

Enabling IPv6

The switch's implementation of IPv6 is disabled by default. To enable IPv6 forwarding, use the ipv6 forwarding command on page 27.8.

To display information about IPv6 settings, use the show ipv6 interface brief command on page 27.27.

Because the switch implements IPv6 as a dual stack, implementing IPv6 does not affect IPv4 functionality.

Note IPv6 is only supported in stand-alone mode. IPv6 is not supported on x510 series switches in VCStack configurations.

IPv6 Stateless Address Autoconfiguration (SLAAC)

The switch's implementation of IPv6 supports SLAAC on an interface. To enable IPv6 SLAAC on an interface, use the ipv6 address autoconfig command on page 27.5. SLAAC automatically applies the MAC address of the interface to an IPv6 address for the interface specified. ipv6 address autoconfig enables automatic configuration of IPv6 addresses on an interface using stateless autoconfiguration, and enables IPv6 processing on an interface.

IPv6 EUI-64 Addressing

The switch's implementation of IPv6 supports EUI-64. To enable IPv6 EUI-64, use the ipv6 address command on page 27.3 specifying the optional **eui64** parameter for an interface. EUI-64 applies the MAC address of the interface to an IPv6 address for the interface. The EUI-64 identifiers from the MAC address are used as the least significant 64 bits of a unicast address.

IPv6 Link-local Addresses

The switch's implementation of IPv6 supports IPv6 link-local addresses without global addresses for communications within the local subnetwork. Routers do not forward packets to link-local addresses. To enable IPv6 link-local addresses, use the ipv6 enable command on page 27.7. ipv6 enable automatically configures an IPv6 link-local address on the interface and enables IPv6 processing on the interface.

Note that link-local addresses are retained in the system until they are negated by using the no variant of the command that established them. See the Link Local Addresses glossary entry, and the ipv6 enable command for more information. Also note that the link-local address is retained in the system if the global address is removed using another command, which was not used to establish the link-local address. For example, if a link local address is established with the ipv6 enable command then it will not be removed using a no ipv6 address command.

IPv6 RA Guard

This section describes the switch's support for IPv6 RA Guard, and how to configure IPv6 RA Guard on the switch.

RA Guard Introduction

Router Advertisements (RAs) and Router Redirects are used to manage IPv6 networks. RA messages advertise a router's presence and specify network parameters that are used by hosts as part of address auto-configuration and setting next-hop routes for particular destinations.

RAs are periodically transmitted by routers allowing networks to be reconfigured by changes to the routers only. Routers can also send redirects to hosts suggesting that they use a different next-hop route for a particular traffic stream. But because the entire network configuration can be modified by what is contained in RAs and redirects, the network is vulnerable to rogue messages that are generated either through misconfiguration or due to a malicious attack.

RA Guard on the switch simply considers each of its ports as either trusted or untrusted. Any host connected to a port is considered trusted or untrusted depending on the port status. A trusted port will accept RAs and redirects and will forward RAs and redirects on trusted ports. An untrusted port will block and discard all RAs and redirects received from the untrusted host.

Enabling IPv6 RA Guard

The switch's implementation of IPv6 RA Guard is disabled by default. To enable IPv6 RA Guard on a port to block RAs from an untrusted host, use the **ipv6 nd raguard** command on page 27.17. Disable IPv6 RA Guard to allow RAs on a port using the **no ipv6 nd raguard** command.

Chapter 27: IPv6 Commands

Command List

This chapter provides an alphabetical reference of commands used to configure IPv6. For more information, see Chapter 26, IPv6 Introduction.

Note IPv6 is only supported in the stand-alone mode. IPv6 is not supported in VCStack configurations.

clear ipv6 neighbors

Use this command to clear all dynamic IPv6 neighbor entries.

Note This command is only supported in the stand-alone mode. IPv6 is not supported in VCStack configurations.

Syntax

```
clear ipv6 neigbors
```

Mode Privileged Exec

Example

```
awplus# clear ipv6 neighbors
```

ipv6 address

Use this command to set the IPv6 address of a VLAN interface and enable IPv6. Use the optional **eui64** parameter to derive the interface identifier of the IPv6 address from the MAC address of the interface. Note that the MAC address of the default VLAN is applied if the interface does not have a MAC address of its own when specifying the **eui64** parameter.

Use the **no** variant of this command to remove the IPv6 address assigned and disable IPv6. Note that if no global addresses are left after removing the IPv6 address then IPv6 is disabled.

Note IPv6 is only supported in the stand-alone mode. IPv6 is not supported in VCStack configurations.

Syntax

```
ipv6 address <ipv6-addr/prefix-length> [eui64]

no ipv6 address <ipv6-addr/prefix-length> [eui64]
```

Parameter	Description
`<ipv6-addr/prefix-length>`	Specifies the IPv6 address to be set, for example, 2001:db8::1/128. The IPv6 address uses the format X:X::X:X/Prefix-Length. The prefix-length is usually set between 0 and 64, or has the value 128.
`[eui64]`	Specifies the lower 64 bits of the IPv6 address from the eui64 interface identifier (EUI - Extended Unique Identifier). EUI-64 identifiers are used as the least significant 64 bits of a unicast network address or a link-local address using stateless autoconfiguration.

Mode Interface Configuration for a VLAN interface.

Usage If the **eui64** parameter is specified then the lower 64 bits of the IPv6 address are replaced with the same address that would be acquired through stateless address autoconfiguration (SLAAC) if the device received an RA (Router Advertisement) specifying this prefix. See ipv6 address autoconfig for a detailed command description and examples to enable and disable SLAAC.

Note that link-local addresses are retained in the system until they are negated by using the no variant of the command that established them. See the Link Local Addresses glossary entry, and the ipv6 enable command for more information.

Also note that the link-local address is retained in the system if the global address is removed using another command, which was not used to establish the link-local address. For example, if a link local address is established with the ipv6 enable command then it will not be removed using a no ipv6 address command.

Example To assign the IPv6 address `2001:0db8::a2/64` to the interface `vlan2`, use the following commands:

```
        awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ipv6 address 2001:0db8::a2/64
```

To remove the IPv6 address `2001:0db8::a2/64` from the interface `vlan2`, use the following commands:

```
        awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# no ipv6 address 2001:0db8::a2/64
```

To assign the **eui64** derived address in the prefix `2001:db8::/48` to interface `vlan2`, use the following commands:

```
             awplus# configure terminal
     awplus(config)# interface vlan2
awplus(config-fr-subif)# ipv6 address 2001:0db8::/48 eui64
```

To remove the **eui64** derived address in the prefix `2001:db8::/48` from interface `vlan2`, use the following commands:

```
             awplus# configure terminal
     awplus(config)# interface vlan2
awplus(config-fr-subif)# no ipv6 address 2001:0db8::/48 eui64
```

Validation Commands show running-config
show ipv6 interface brief
show ipv6 route

Related Commands ipv6 address autoconfig

ipv6 address autoconfig

Use this command to enable IPv6 stateless address autoconfiguration (SLAAC) for an interface. This configures an IPv6 address on an interface derived from the MAC address on the interface.

Use the **no** variant of this command to disable IPv6 SLAAC on an interface. Note that if no global addresses are left after removing all IPv6 autoconfigured addresses then IPv6 is disabled.

Note This command is only supported in the stand-alone mode. IPv6 is not supported in VCStack configurations.

Syntax

```
ipv6 address autoconfig

no ipv6 address autoconfig
```

Mode Interface Configuration for a VLAN interface.

Usage The ipv6 address autoconfig command enables automatic configuration of IPv6 addresses using stateless autoconfiguration on an interface and enables IPv6, but does not enable IPv6 forwarding. See ipv6 forwarding command on page 27.8 for further description and examples.

IPv6 hosts can configure themselves when connected to an IPv6 network using ICMPv6 (Internet Control Message Protocol version 6) router discovery messages. Configured routers respond with a Router Advertisement (RA) containing configuration parameters for IPv6 hosts.

The SLAAC process derives the interface identifier of the IPv6 address from the MAC address of the interface. When applying SLAAC to an interface, note that the MAC address of the default VLAN is applied to the interface if the interface does not have its own MAC address.

If SLAAC is not suitable then a network can use stateful configuration with DHCPv6 (Dynamic Host Configuration Protocol version 6) Relay, or hosts can be configured statically. See ip dhcp-relay server-address for the DHCPv6 Relay server command description and examples. For introduction and configuration information about DHCPv6 Relay agent see "DHCP Relay Agent Introduction" on page 60.8 and "Configuring the DHCP Relay Agent" on page 60.8.

Note that link-local addresses are retained in the system until they are negated by using the no variant of the command that established them. See the Link Local Addresses glossary entry, and the ipv6 enable command for more information.

Also note that the link-local address is retained in the system if the global address is removed using another command, which was not used to establish the link-local address. For example, if a link local address is established with the ipv6 enable command then it will not be removed using a no ipv6 address command.

Example To enable SLAAC on the interface `vlan2`, use the following commands:

```
         awplus# configure terminal
 awplus(config)# interface vlan2
awplus(config-if)# ipv6 address autoconfig
```

To disable SLAAC on the interface `vlan2`, use the following commands:

```
         awplus# configure terminal
 awplus(config)# interface vlan2
awplus(config-if)# no ipv6 address autoconfig
```

Validation Commands

show running-config
show ipv6 interface brief
show ipv6 route

Related Commands

ipv6 address
ipv6 enable

ipv6 enable

Use this command to enable IPv6 on an interface without an IPv6 global address for the interface. This enables IPv6 with a IPv6 link-local address, not an IPv6 global address.

Use the no variant of this command to disable IPv6 on an interface without a global address. Note the no variant of this command does not operate on an interface with an IPv6 global address or an interface configured for IPv6 stateless address autoconfiguration (SLAAC),

Syntax

```
ipv6 enable

no ipv6 enable
```

Mode Interface Configuration

Note This command is only supported in the stand-alone mode. IPv6 is not supported in VCStack configurations.

Usage The ipv6 enable command automatically configures an IPv6 link-local address on the interface and enables the interface for IPv6 processing. Note that link-local addresses are retained in the system until they are negated by using the no variant of the command that established them. See the Link Local Addresses glossary entry for more information.

Also note that the link-local address is retained in the system if the global address is removed using another command, which was not used to establish the link-local address. For example, if a link local address is established with the ipv6 enable command then it will not be removed using a no ipv6 address command.

Example To enable IPv6 with only a link-local IPv6 address on the interface vlan2, use the following commands:

```
         awplus# configure terminal
 awplus(config)# interface vlan2
awplus(config-if)# ipv6 enable
```

To disable IPv6 with only a link-local IPv6 address on the interface vlan2, use the following commands:

```
         awplus# configure terminal
 awplus(config)# interface vlan2
awplus(config-if)# no ipv6 enable
```

Validation Commands

show running-config
show ipv6 interface brief
show ipv6 route

Related Commands

ipv6 address
ipv6 address autoconfig

ipv6 forwarding

Use this command to turn on IPv6 forwarding.

Syntax `ipv6 forwarding`

Mode Global Configuration

Usage Note that there is no accompanying **"no ipv6 forwarding**" command. The default mode is ipv6 forwarding - OFF. However; If ipv6 forwarding has been turned on, you can use the following procedure to turn it off:

- Run the show boot command to display the current config file.
- Delete the line that contains the text, "ipv6 forwarding."
- Reboot the switch.

Note This command is only supported in the stand-alone mode. IPv6 is not supported in VCStack configurations.

Example

```
        awplus# configure terminal
awplus(config)# ipv6 forwarding
```

ipv6 nd managed-config-flag

Use this command to set the managed address configuration flag, contained within the router advertisement field.

Setting this flag indicates the operation of a stateful autoconfiguration protocol such as DHCPv6 for address autoconfiguration, and that address information (i.e. the network prefix) and other (non-address) information can be requested from the switch.

An unset flag enables hosts receiving the advertisements to use a sateless autoconfiguration mechanism to establish their IPv6 addresses. The default is flag unset.

Use the **no** variant of this command to reset this command to its default of, *flag unset*.

Note This command is only supported in the stand-alone mode. IPv6 is not supported in VCStack configurations.

Syntax

```
ipv6 nd managed-config-flag

no ipv6 nd managed-config-flag
```

Default Unset

Mode Interface Configuration for a VLAN interface.

Usage Advertisement flags will not be transmitted unless you have applied the no ipv6 nd suppress-ra command on page 27.21. This step is included in the example below.

Example To set the managed address configuration flag on `vlan2`, use the following commands:

```
          awplus#  configure terminal
  awplus(config)#  interface vlan2
awplus(config-if)# ipv6 nd managed-config-flag
awplus(config-if)# no ipv6 nd suppress-ra
```

Related Commands

ipv6 nd suppress-ra
ipv6 nd prefix
ipv6 nd other-config-flag

ipv6 nd minimum-ra-interval

Use this command in Interface Configuration mode to set a minimum Router Advertisement (RA) interval for a VLAN interface.

Use the **no** variant of this command in Interface Configuration mode to remove the minimum RA interval for a VLAN interface.

Note This command is only supported in the stand-alone mode. IPv6 is not supported on x510 series switches in VCStack configurations.

Syntax

```
ipv6 nd minimum-ra-interval <seconds>

no ipv6 nd minimum-ra-interval [<seconds>]
```

Parameter	Description
<seconds>	Specifies the number of seconds between IPv6 Router Advertisements (RAs). Valid values are from 3 to 1350 seconds.

Default The RA interval for a VLAN interface is unset by default.

Mode Interface Configuration for a VLAN interface.

Examples To set the minimum RA interval for VLAN interface `vlan2`, use the following commands:

```
          awplus# configure terminal
  awplus(config)# interface vlan2
awplus(config-if)# ipv6 nd minimum-ra-interval 60
```

To remove the minimum RA interval for VLAN interface `vlan2`, use the following commands:

```
          awplus# configure terminal
  awplus(config)# interface vlan2
awplus(config-if)# no ipv6 nd minimum-ra-interval 60
```

Related Commands

ipv6 nd ra-interval
ipv6 nd suppress-ra
ipv6 nd prefix
ipv6 nd other-config-flag

ipv6 nd other-config-flag

Use this command to set the **other** stateful configuration flag (contained within the router advertisement field) to be used for IPv6 address auto-configuration. This flag is used to request the router to provide information in addition to providing addresses.

Note Setting the ipv6 nd managed-config-flag command on page 27.9 implies that the ipv6 nd other-config-flag will also be set.

Use **no** variant of this command to reset the value to the default.

Note This command is only supported in the stand-alone mode. IPv6 is not supported in VCStack configurations.

Syntax

```
ipv6 nd other-config-flag
```

```
no ipv6 nd other-config-flag
```

Default Unset

Mode Interface Configuration for a VLAN interface.

Usage Advertisement flags will not be transmitted unless you have applied the no ipv6 nd suppress-ra command on page 27.21. This step is included in the example below.

Example To set the ipv6 other-config-flag on `vlan4`, use the following commands:

```
          awplus# configure terminal
  awplus(config)# interface vlan4
awplus(config-if)# ipv6 nd other-config-flag
awplus(config-if)# no ipv6 nd suppress-ra
```

Related Commands

ipv6 nd suppress-ra
ipv6 nd prefix
ipv6 nd managed-config-flag

ipv6 nd prefix

Use this command in Interface Configuration mode for a VLAN interface to specify the IPv6 prefix information that is advertised by the router advertisement for IPv6 address auto-configuration.

Use the **no** parameter with this command to reset the IPv6 prefix for a VLAN interface in Interface Configuration mode.

Note This command is only supported in the stand-alone mode. IPv6 is not supported in VCStack configurations.

Syntax

```
ipv6 nd prefix <ipv6-prefix/length>

ipv6 nd prefix <ipv6-prefix/length> [<valid-lifetime>]

ipv6 nd prefix <ipv6-prefix/length> <valid-lifetime>
    <preferred-lifetime> [no-autoconfig]

ipv6 nd prefix <ipv6-prefix/length> <valid-lifetime>
    <preferred-lifetime> off-link [no-autoconfig]

no ipv6 nd prefix [<ipv6-addr/prefix-length>|all]
```

Parameter	Description
<ipv6-prefix/length>	The prefix to be advertised by the router. The IPv6 address prefix uses the format X:X::/prefix-length. The prefix-length is usually set between 0 and 64, or has the value 128.The default is X:X::/64.
<valid-lifetime>	The the period during which the specified IPv6 address prefix is valid. This can be set to a value between 0 and 4294967295 seconds. The default is 2592000 (30 days). Note that this period should be set to a value greater than that set for the prefix preferred-lifetime.
<preferred-lifetime>	Specifies theIPv6 prefix preferred lifetime. This is the period during which the IPv6 address prefix is considered a current (undeprecated) value. After this period, the command is still valid but should not be used in new communications. Set to a value between 0 and 4294967295 seconds. The default is 604800 seconds (7 days). Note that this period should be set to a value less than that set for the prefix valid-lifetime.
`off-link`	Specify the IPv6 prefix off-link flag. The default is *flag set*.
`no-autoconfig`	Specify the IPv6 prefix no autoconfiguration flag. Setting this flag indicates that the prefix is not to be used for autoconfiguration. The default is *flag set*.
`all`	Specify all IPv6 prefixes associated with the VLAN interface.

Default Valid-lifetime default is 2592000 seconds (30 days). Preferred-lifetime default is 604800 seconds (7 days).

Mode Interface Configuration for a VLAN interface.

Usage This command specifies the IPv6 prefix flags that are advertised by the router advertisement message.

Examples The following example configures the switch to issue router advertisements on VLAN interface `vlan4`, and advertises the address prefix of `2001:0db8::/64`.

```
          awplus# configure terminal
  awplus(config)# interface vlan4
awplus(config-if)# ipv6 nd prefix 2001:0db8::/64
```

The following example configures the switch to issue router advertisements on VLAN interface `vlan4`, and advertises the address prefix of `2001:0db8::/64` with a valid lifetime of 10 days and a preferred lifetime of 5 days.

```
          awplus# configure terminal
  awplus(config)# interface vlan4
awplus(config-if)# ipv6 nd prefix 2001:0db8::/64 864000 432000
```

The following example configures the switch to issue router advertisements on VLAN interface `vlan4`, and advertises the address prefix of `2001:0db8::/64` with a valid lifetime of 10 days, a preferred lifetime of 5 days and no prefix used for autoconfiguration.

```
          awplus# configure terminal
  awplus(config)# interface vlan4
awplus(config-if)# ipv6 nd prefix 2001:0db8::/64 864000 43200
                   no-autoconfig
```

The following example resets router advertisements on VLAN interface `vlan4`, so the address prefix of `2001:0db8::/64` is not advertised from the switch.

```
          awplus#  configure terminal
  awplus(config)#  interface vlan4
awplus(config-if)# no ipv6 nd prefix 2001:0db8::/64
```

The following example resets all router advertisements on VLAN interface `vlan4`:

```
          awplus#  configure terminal
  awplus(config)#  interface vlan4
awplus(config-if)# no ipv6 nd prefix all
```

Related Commands ipv6 nd suppress-ra

ipv6 nd ra-interval

Use this command to specify the interval between IPv6 Router Advertisements (RA) transmissions.

Use **no** parameter with this command to reset the value to the default value (600 seconds).

Note This command is only supported in the stand-alone mode. IPv6 is not supported in VCStack configurations.

Syntax

```
ipv6 nd ra-interval <seconds>

no ipv6 nd ra-interval
```

Parameter	Description
<seconds>	Specifies the number of seconds between IPv6 Router Advertisements (RAs). Valid values are from 4 to 1800 seconds.

Default 600 seconds.

Mode Interface Configuration for a VLAN interface.

Usage Advertisement flags will not be transmitted unless you have applied the no ipv6 nd suppress-ra command on page 27.21 as shown in the example below.

Example To set the advertisements interval on `vlan4` to be 60 seconds, use the following commands:

```
           awplus# configure terminal
   awplus(config)# interface vlan4
awplus(config-if)# ipv6 nd ra-interval 60
awplus(config-if)# no ipv6 nd suppress-ra
```

Related Commands ipv6 nd minimum-ra-interval
ipv6 nd suppress-ra
ipv6 nd prefix

ipv6 nd ra-lifetime

Use this command to specify the time period that this router can usefully act as a default gateway for the network. Each router advertisement resets this time period.

Use **no** parameter with this command to reset the value to default.

Note This command is only supported in the stand-alone mode. IPv6 is not supported in VCStack configurations.

Syntax

```
ipv6 nd ra-lifetime <seconds>

no ipv6 nd ra-lifetime
```

Parameter	Description
<seconds>	Time period in seconds. Valid values are from 0 to 9000. Note that you should set this time period to a value greater than the value you have set using the ipv6 nd ra-interval command.

Default 1800 seconds

Mode Interface Configuration for a VLAN interface.

Usage This command specifies the lifetime of the current router to be announced in IPv6 Router Advertisements.

Advertisement flags will not be transmitted unless you have applied the no ipv6 nd suppress-ra command. This instruction is included in the example shown below.

Example To set the advertisement lifetime of 8000 seconds on `vlan4`, use the following commands:

```
          awplus#  configure terminal
  awplus(config)#  interface vlan4
awplus(config-if)#  ipv6 nd ra-lifetime 8000
awplus(config-if)#  no ipv6 nd suppress-ra
```

Related Commands ipv6 nd suppress-ra
ipv6 nd prefix

ipv6 nd raguard

Use this command to apply the Router Advertisements (RA) Guard feature from the Interface Configuration mode for a switch port. This blocks all RA messages received on a switch port. For introductory information about RA Guard see "RA Guard Introduction" on page 26.11. Use the **no** parameter with this command to disable RA Guard for a specified switch port.

Note This command is only supported in the stand-alone mode. IPv6 is not supported in VCStack configurations.

Syntax

```
ipv6 nd raguard
```

```
no ipv6 nd raguard
```

Default RA Guard is not enabled by default.

Mode Interface Configuration for a switch port interface.

Usage Router Advertisements (RAs) are used by Routers to announce themselves on the link. Applying RA Guard to a switch port disallows Router Advertisements and redirect messages. RA Guard blocks RAs from untrusted hosts. Blocking RAs stops untrusted hosts from flooding malicious RAs and stops any misconfigured hosts from disrupting traffic on the local network.

Enabling RA Guard on a port blocks RAs from a connected host and indicates the port and host are untrusted. Disabling RA Guard on a port allows RAs from a connected host and indicates the port and host are trusted. Ports and hosts are trusted by default to allow RAs.

Example To enable RA Guard on switch ports `port1.0.2-1.0.12`, use the following commands:

```
         awplus#  configure terminal
 awplus(config)#  interface port1.0.2-1.0.12
awplus(config-if)#  ipv6 nd raguard
```

To verify RA Guard is enabled on switch port interface `port1.0.2`, use the command:

```
awplus#  show running-config interface port1.0.2
```

Output Example output from a **show running-config interface port1.0.2** to verify RA Guard:

```
!
interface port1.0.2
 switchport mode access
 ipv6 nd raguard
!
```

To disable RA Guard on switch ports `port1.0.2-1.0.12`, use the following commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.2-port1.0.12
awplus(config-if)# no ipv6 nd raguard
```

When RA Guard is disabled on a switch port it is not displayed in **show running-config** output.

Related Commands show running-config interface

ipv6 nd reachable-time

Use this command to specify the reachable time in the router advertisement to be used for detecting reachability of the IPv6 neighbor.

Use the **no** variant of this command to reset the value to default.

Note This command is only supported in the stand-alone mode. IPv6 is not supported in VCStack configurations.

Syntax

```
ipv6 nd reachable-time <milliseconds>
```

```
no ipv6 nd reachable-time
```

Parameter	Description
<milliseconds>	Time period in milliseconds. Valid values are from 1000 to 3600000. Setting this value to 0 indicates an unspecified reachable-time.

Default 0 milliseconds

Mode Interface Configuration for a VLAN interface.

Usage This command specifies the reachable time of the current router to be announced in IPv6 Router Advertisements.

Advertisement flags will not be transmitted unless you have applied the no ipv6 nd suppress-ra command. This instruction is included in the example shown below.

Example To set the reachable-time in router advertisements on `vlan4` to be 1800000 milliseconds, enter the following commands:

```
          awplus# configure terminal
  awplus(config)# interface vlan4
awplus(config-if)# ipv6 nd reachable-time 1800000
awplus(config-if)# no ipv6 nd suppress-ra
```

To reset the reachable-time in router advertisements on `vlan4` to an unspecified reachable-time (0 milliseconds), enter the following commands:

```
          awplus# configure terminal
  awplus(config)# interface vlan4
awplus(config-if)# no ipv6 nd reachable-time
```

Related Commands ipv6 nd suppress-ra
ipv6 nd prefix

ipv6 nd retransmission-time

Use this command to specify the advertised retransmission interval for Neighbor Solicitation in milliseconds between IPv6 Routers.

Use the **no** variant of this command to reset the retransmission time to the default (1 second).

Note This command is only supported in the stand-alone mode. IPv6 is not supported in VCStack configurations.

Syntax

```
ipv6 nd retransmission-time <milliseconds>

no ipv6 nd retransmission-time [<milliseconds>]
```

Parameter	Description
`<milliseconds>`	Time period in milliseconds. Valid values are from 1000 to 3600000.

Default 1000 milliseconds (1 second)

Mode Interface Configuration for a VLAN interface.

Example To set the retransmission-time of Neighbor Solicitation on `vlan2` to be 800000 milliseconds, enter the following commands:

```
          awplus# configure terminal
  awplus(config)# interface vlan2
awplus(config-if)# ipv6 nd retransmission-time 800000
```

To reset the retransmission-time of Neighbor Solicitation on `vlan2` to the default 1000 milliseconds (1 second), enter the following commands:

```
          awplus# configure terminal
  awplus(config)# interface vlan2
awplus(config-if)# no ipv6 nd retransmission-time
```

Related Commands ipv6 nd suppress-ra
ipv6 nd prefix

ipv6 nd suppress-ra

Use this command to inhibit IPv6 Router Advertisement (RA) transmission for the current interface. Router advertisements are used when applying IPv6 stateless auto-configuration.

Use **no** parameter with this command to enable Router Advertisement transmission.

Note This command is only supported in the stand-alone mode. IPv6 is not supported in VCStack configurations.

Syntax

```
ipv6 nd suppress-ra
```

```
no ipv6 nd suppress-ra
```

Default Router Advertisement (RA) transmission is suppressed by default.

Mode Interface Configuration for a VLAN interface.

Example To enable the transmission of router advertisements from interface `vlan4` on the switch, use the following commands:

```
        awplus# configure terminal
awplus(config)# interface vlan4
awplus(config-if)# no ipv6 nd suppress-ra
```

Related Commands
ipv6 nd ra-interval
ipv6 nd prefix

ipv6 neighbor

Use this command to add a static IPv6 neighbor entry.

Use the **no** variant of this command to remove a specific IPv6 neighbor entry.

Note This command is only supported in the stand-alone mode. IPv6 is not supported in VCStack configurations.

Syntax

```
ipv6 neighbor <ipv6-address> <vlan-name> <mac-address> <port-list>

no ipv6 neighbor <ipv6-address> <vlan-name> <port-list>
```

Parameter	Description
<ipv6-address>	Specify the neighbor's IPv6 address in format `X:X::X:X`.
<vlan-name>	Specify the neighbor's VLAN name.
<mac-address>	Specify the MAC hardware address in hexadecimal notation with the format `HHHH.HHHH.HHHH`.
<port-list>	Specify the port number, or port range.

Mode Global Configuration

Usage Use this command to clear a specific IPv6 neighbor entry. To clear all dynamic address entries, use the clear ipv6 neighbors command.

Example To create a static neighbor entry for IPv6 address `2001:0db8::a2, on vlan 4,` MAC address `0000.cd28.0880, on port1.0.19,` use the command:

```
        awplus# configure terminal

awplus(config)# ipv6 neighbor 2001:0db8::a2 vlan4
                0000.cd28.0880 port1.0.19
```

Related Commands clear ipv6 neighbors

ipv6 opportunistic-nd

Use this command to enable opportunistic neighbor discovery for the global IPv6 ARP cache. Opportunistic neighbor discovery changes the behavior for unsolicited ARP packet forwarding on the switch.

Use the **no** variant of this command to disable opportunistic neighbor discovery for the global IPv6 ARP cache.

Note This command is only supported in the stand-alone mode. IPv6 is not supported in VCStack configurations.

Syntax

```
ipv6 opportunistic-nd

no ipv6 opportunistic-nd
```

Default Opportunistic neighbor discovery is disabled by default.

Mode Global Configuration

Usage When opportunistic neighbor discovery is enabled, the switch will reply to any received unsolicited ARP packets (but not gratuitous ARP packets). The source MAC address for the unsolicited ARP packet is added to the IPv6 ARP cache, so the switch forwards the ARP packet. When opportunistic neighbor discovery is disabled, the source MAC address for the ARP packet is not added to the IPv6 ARP cache, so the ARP packet is not forwarded by the switch.

Use the arp opportunistic-nd command to enable opportunistic neighbor discovery in Global Configuration mode for all interfaces on the switch configured for IPv4. Use a show arp command to confirm opportunistic neighbor discovery is configured on the switch.

Example To enable opportunistic neighbor discovery for the IPv6 ARP cache, enter:

```
        awplus#  configure terminal
awplus(config)#  ipv6 opportunistic-nd
```

To disable opportunistic neighbor discovery for the IPv6 ARP cache, enter:

```
        awplus#  configure terminal
awplus(config)#  no ipv6 opportunistic-nd
```

Related Commands arp opportunistic-nd

Validation Commands show arp

ipv6 route

Use this command to establish the distance for static routes of a network prefix.

Use the **no** variant of this command to disable the distance for static routes of the network prefix.

Note This command is only supported in the stand-alone mode. IPv6 is not supported in VCStack configurations.

Syntax

```
ipv6 route <dest-prefix> <dest-prefix/length>
   {<gateway-ip>|<gateway-name>} [<distvalue>]

no ipv6 route <dest-prefix> <dest-prefix/length>
   {<gateway-ip>|<gateway-name>} [<distvalue>]
```

Parameter	Description
<dest-prefix/ length>	Specifies the IP destination prefix. The IPv6 address prefix uses the format X:X::/prefix-length. The prefix-length is usually set between 0 and 64, or has the value 128.
<gateway-ip>	Specifies the IP gateway (or next hop) address. The IPv6 address uses the format X:X::X:X/Prefix-Length. The prefix-length is usually set between 0 and 64, or has the value 128.
<distvalue>	Specifies the administrative distance for the route. Valid values are from 1 to 255.
<gateway-name>	Specifies the name of the gateway (or next hop) interface.

Mode Global Configuration

Example

```
        awplus# configure terminal
awplus(config)# ipv6 route 2001:0db8::1/128 myintname 32
```

Validation Commands show running-config
show ipv6 route

ping ipv6

This command sends a query to another IPv6 host (send Echo Request messages).

Syntax

```
ping ipv6 [<host>|<ipv6-address>] [repeat {<1-2147483647>|
    continuous}] [size <10-1452>] [interface <interface-list>]
    [timeout <1-65535>]
```

Parameter	Description
<ipv6-addr>	The destination IPv6 address. The IPv6 address uses the format X:X::X:X.
<hostname>	The destination hostname.
`repeat`	Specify the number of ping packets to send.
<1-2147483647>	Specify repeat count. The default is 5.
`size` *<10-1452>*	The number of data bytes to send, excluding the 8 byte ICMP header. The default is 56 (64 ICMP data bytes).
`interface` *<interface-list>*	The interface or range of configured IP interfaces to use as the source in the IP header of the ping packet.
`timeout` *<1-65535>*	The time in seconds to wait for echo replies if the ARP entry is present, before reporting that no reply was received. If no ARP entry is present, it does not wait.
`repeat`	Specify the number of ping packets to send.
<1-2147483647>	Specify repeat count. The default is 5.
`continuous`	Continuous ping.
`size` *<10-1452>*	The number of data bytes to send, excluding the 8 byte ICMP header. The default is 56 (64 ICMP data bytes).
`timeout` *<1-65535>*	The time in seconds to wait for echo replies if the ARP entry is present, before reporting that no reply was received. If no ARP entry is present, it does not wait.

Mode User Exec and Privileged Exec

Example

```
awplus# ping ipv6 2001:0db8::a2
```

Related Commands traceroute ipv6

show ipv6 forwarding

Use this command to display IPv6 forwarding status.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax

```
show ipv6 forwarding
```

Mode User Exec and Privileged Exec

Example

```
awplus# show ipv6 forwarding
```

Output Figure 27-1: Example output from the **show ipv6 forwarding** command

```
ipv6 forwarding is on
```

show ipv6 interface brief

Use this command to display brief information about interfaces and the IPv6 address assigned to them, and optionally Neighbor Discovery configurations for a configured interface.

To display information about a specific interface, specify the interface name with the command.

For information on output options, see “Controlling “show” Command Output” on page 1.41.

Note This command is only supported in the stand-alone mode. IPv6 is not supported in VCStack configurations.

Syntax

```
show ipv6 interface [brief|<interface> [nd]]
```

Parameter	Description
`brief`	Specify this optional parameter to display brief IPv6 interface information.
`<interface>`	Specify the configured interface to display IPv6 information about. For instance `vlan2`. The specified interface must exist and be configured.
`nd`	Specify this optional parameter for Neighbor Discovery configurations.

Mode User Exec and Privileged Exec

Examples

```
awplus# show ipv6 interface brief
```

```
awplus# show ipv6 interface brief vlan1 nd
```

Output Figure 27-2: Example output from the **show ipv6 interface brief** command

```
Interface     IPv6-Address                                Status       Protocol
lo            unassigned                                  admin up     running
vlan1         unassigned                                  admin up     running
vlan10        2001:db8:a:0:0:c0a8:a0b/64                  admin up     running
              2001:db8::200:cdff:fe28:84a/64
vlan20        2001:db8::14:0:0:c0a8:140b/64               admin up     running
              2001:db8::200:cdff:fe28:84a/64
vlan30        2001:db8::1e:0:0:c0a8:1e0b/64               admin up     running
              2001:db8::200:cdff:fe28:84a/64
vlan40        2001:db8::28:0:0:c0a8:280b/64               admin up     running
              2001:db8::200:cdff:fe28:84a/64
vlan201       unassigned                                  admin up     running
vlan250       2001:db8::fa:0:0:c0a8:fa0b/64               admin up     running
              2001:db8::200:cdff:fe28:84a/64
```

Related Commands show interface brief

show ipv6 neighbors

Use this command to display all IPv6 neighbors.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Note This command is only supported in the stand-alone mode. IPv6 is not supported in VCStack configurations.

Syntax

```
show ipv6 neighbors
```

Mode User Exec and Privileged Exec

show ipv6 route

Use this command to display the IPv6 routing table for a protocol or from a particular table.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Note This command is only supported in the stand-alone mode. IPv6 is not supported in VCStack configurations.

Syntax

```
show ipv6 route
    [connected|database|static|summary|
    <ipv6-address>|<ipv6-addr/prefix-length>)]
```

Parameter	Description
`connected`	Displays only the routes learned from connected interfaces.
`database`	Displays only the IPv6 routing information extracted from the database.
`static`	Displays only the IPv6 static routes you have configured.
`summary`	Displays summary information from the IPv6 routing table.
<ipv6-address>	Displays the routes for the specified address in the IP routing table. The IPv6 address uses the format X:X::X:X/Prefix-Length. The prefix-length is usually set between 0 and 64, or has the value 128.
<ipv6-prefix/length>	Displays only the routes for the specified IP prefix. The IPv6 address prefix uses the format X:X::/prefix-length. The prefix-length is usually set between 0 and 64, or has the value 128.

Mode User Exec and Privileged Exec

Example 1

```
awplus# show ipv6 route
```

Example 2

```
awplus# show ipv6 route database
```

Output

Figure 27-3: Example output of the **show ipv6 route database** command

```
IPv6 Routing Table
Codes: C - connected, S - static,
       > - selected route, * - FIB route, p - stale info
Timers: Uptime

S    ::/0 [1/0] via 2001::a:0:0:c0a8:a01 inactive, 6d22h12m
          [1/0] via 2001::fa:0:0:c0a8:fa01 inactive, 6d22h12m
```

show ipv6 route summary

Use this command to display the summary of the current NSM RIB entries.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Note This command is only supported in the stand-alone mode. IPv6 is not supported in VCStack configurations.

Syntax

```
show ipv6 route summary
```

Mode User Exec and Privileged Exec

Example

```
awplus# show ipv6 route summary
```

Output Figure 27-4: Example output from the **show ipv6 route summary** command

```
IPv6 routing table name is Default-IPv6-Routing-Table(0)
IPv6 routing table maximum-paths is 4
RouteSource     Networks
connected       4
FIB             5
```

Related Commands show ip route
show ip route database

traceroute ipv6

Use this command to trace the route to the specified IPv6 host.

Syntax

```
traceroute ipv6 {<ipv6-addr>|<hostname>}
```

Parameter	Description
<ipv6-addr>	The destination IPv6 address. The IPv6 address uses the format X:X::X:X.
<hostname>	The destination hostname.

Mode User Exec and Privileged Exec

Example

```
awplus# traceroute ipv6 2001:0db8::a2
```

Related Commands ping ipv6

Part 4: Multicast Applications

Chapter 28: IGMP and IGMP Snooping Introduction

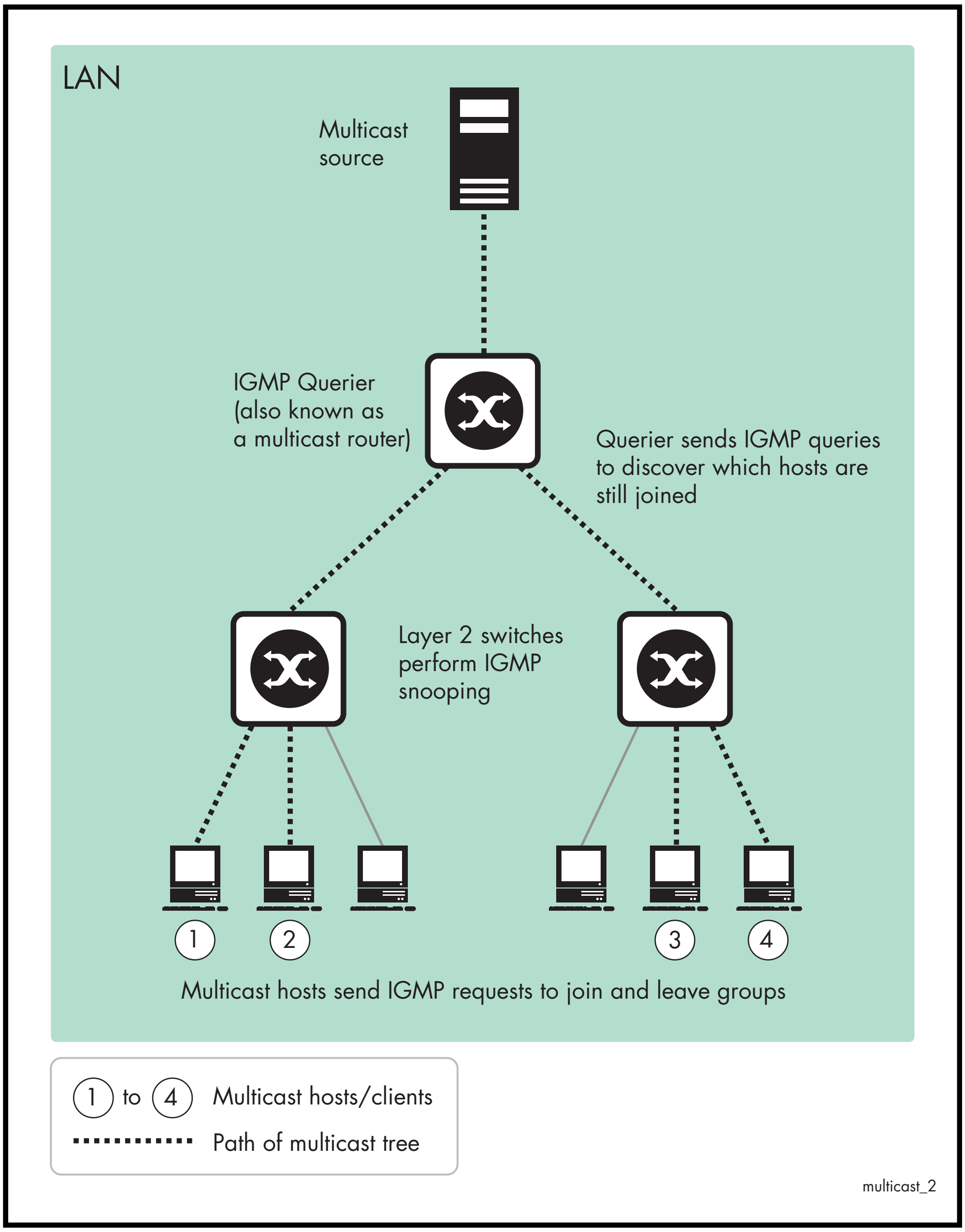
Introduction

This chapter provides information about Internet Group Management Protocol (IGMP), IGMP Snooping, and an introduction to the Query Solicitation feature when used with IGMP Snooping. To see details on the commands used in this example, or to see the outputs of the validation commands, refer to Chapter 29, IGMP and IGMP Snooping Commands. For a general overview of multicasting, see Chapter 49, Multicast Introduction and Commands.

IGMP

Internet Group Management Protocol (IGMP) is the protocol that hosts use to indicate that they are interested in receiving a particular multicast stream. An example of a multicast system within a single Layer 2 LAN is shown in Figure 28-1.

Figure 28-1: Multicast system within a single LAN


LAN
Multicast source
IGMP Querier (also known as a multicast router)
Querier sends IGMP queries to discover which hosts are still joined
Layer 2 switches perform IGMP snooping
1
2
3
4
Multicast hosts send IGMP requests to join and leave groups
1 to 4 Multicast hosts/clients
Path of multicast tree
multicast_2

Joining a multicast group (Membership report)

When a host wants to receive a stream, referred to as "joining a group", it sends out an IGMP packet containing the address of the group it wants to join. This packet is called an IGMP Membership report, often referred to as a "join packet". This packet is forwarded through the LAN to the local IGMP querier, which is typically a router. Once the querier has received an IGMP join message, it knows to forward the multicast stream to the host. If it is not already receiving the stream, it must tell the devices between itself and the multicast source, which may be some hops away from the querier, that it wishes to receive the stream. This might involve a process of using Layer 3 multicast protocols to signal across a WAN, or it might be as simple as receiving a stream from a locally connected multicast server.

Staying in the multicast group (Query message)

The Query message is used by a querier to determine whether hosts are still interested in an IGMP group. At certain time intervals (the default is 125 seconds), the querier sends an IGMP query message onto the local LAN. The destination address of the query message is a special "all multicast groups" address. The purpose of this query is to ask "Are there any hosts on the LAN that wish to remain members of multicast groups?" After receiving an IGMP query, any host that wants to remain in a multicast group must send a new join packet for that group. If a host is a member of more than one group, then it sends a join message for each group it wants to remain a member of. The querier looks at the responses it receives to its query, and compares these to the list of multicast streams that it is currently registered to forward. If there are any items in that list for which it has not received query responses, it will stop forwarding those streams. Additionally, if it is receiving those streams through a Layer 3 network, it will send a Layer 3 routing protocol message upstream, asking to no longer receive that stream.

Leaving the multicast group (Leave message)

How a host leaves a group depends on the IGMP version that it is using. Under IGMP version 1, when a host has finished with a data stream, the local querier continues to send the stream to the host until it sends out the next query message and receives no reply back from the host. IGMP version 2 introduced the Leave message. This allows a host to explicitly inform its querier that it wants to leave a particular multicast group. When the querier receives the Leave message, it sends out a group specific query asking whether any hosts still want to remain members of that specific group. If no hosts respond with join messages for that group, then the querier knows that there are no hosts on its LAN that are still members of that group. This means that for that specific group, it can ask to be pruned from the multicast tree. IGMP version 3 removed the Leave message. Instead a host leaves a group by sending a join message with no source specified.

IGMP Snooping

IGMP Snooping is a way for Layer 2 switches to reduce the amount of multicast traffic on a LAN. The AlliedWare Plus implementation of IGMP Snooping is compatible with networks running all IGMP versions.

Without IGMP Snooping, Layer 2 switches handle IP multicast traffic in the same manner as broadcast traffic and forward multicast frames received on one port to all other ports in the same VLAN. IGMP Snooping allows switches to monitor network traffic, and determine hosts to receive multicast traffic, by looking into IGMP packets to learn which attached hosts need to receive which multicast groups. This allows the switch to forward multicast traffic only out the appropriate ports. If it sees multiple reports sent for one group, it will forward only one of them.

How IGMP Snooping operates

IGMP Snooping operates similarly to the multicast protocols. When a switch hears an IGMP report from a host for a given multicast group, the switch adds the host's port number to the list of ports that are listening to the multicast group. When the switch hears an IGMP leave, it removes the host's port from the list, after the completion of the leave process as described in "Leaving the multicast group (Leave message)" on page 28.3. When there are no hosts listening to a group, the switch informs the local querier to stop sending that group's multicast stream.

IGMP Snooping allows query messages to be forwarded to all ports. The hosts that still require the stream respond to the queries by sending reports. The switch intercepts these. Depending on configuration settings, the switch may just forward the reports directly on to the querier, or it may proxy report on behalf of the group, only forwarding on one consolidated report for each group.

By default, IGMP Snooping is enabled both globally and on all VLANs.

Note IGMP Snooping cannot be disabled on an interface if IGMP Snooping has already been disabled globally. IGMP Snooping can be disabled on both an interface and globally if disabled on the interface first and then disabled globally.

To disable IGMP Snooping either		
1.	awplus# `configure terminal`	Enter Global Configuration mode.
2.	awplus(config)# `no ip igmp snooping`	Disable IGMP Snooping globally.
or		
1.	awplus# `configure terminal`	Enter Global Configuration mode.
2.	awplus(config)# `interface <vlan-name>`	Enter Interface Configuration mode for a specific VLAN.
3.	awplus(config-if)# `no ip igmp snooping`	Disable IGMP Snooping for a specific VLAN.

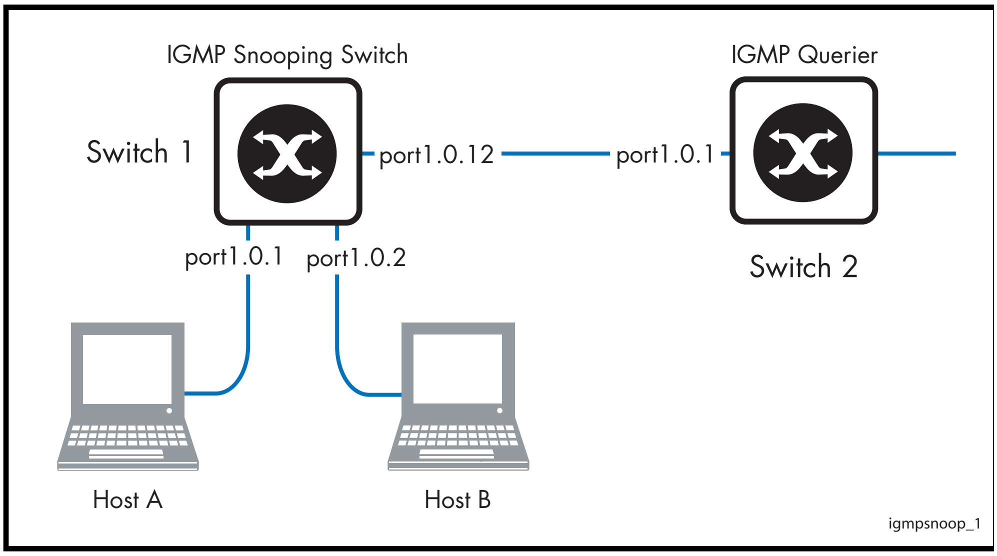
IGMP Snooping and Querier configuration example

This example describes the configuration of IGMP Snooping on an Allied Telesis managed Layer 3 switch (Switch 1) and the configuration of IGMP Querier (Switch 2). The interface port1.0.12 is configured as a multicast router port. Host A and Host B are both members of the same multicast group.

To enable IGMP Snooping on an interface:

- Enable IGMP Snooping globally, if necessary. IGMP Snooping is enabled by default.
- Statically configure ports that are connected to routers if necessary.

Figure 28-2: IGMP Snooping configuration example



As a result of this configuration:

- Membership reports are generated by hosts. The IGMP Snooping switch will forward the membership reports to its router port. Queries received by the IGMP Snooping switch from the IGMP Querier on port1.0.12 are forwarded by the IGMP Snooping switch.
- Because Host A and Host B are members of the same multicast group, the switch does not notify the IP IGMP routing device (IGMP Querier) when Host A leaves the group, because the group still has another member Host B remaining. When Host B also leaves the group, the switch forwards the leave message to the IP IGMP Querier.
- In this example, the configuration of a static mrouter port on port1.0.12 is provided to illustrate the ip igmp snooping mrouter command. However, this command would probably not be necessary, since the switch should dynamically set port1.0.12 to be an mrouter port as it receives IGMP Queries arriving from the IGMP Querier attached to port1.0.12.
- In this example, it is not necessary to explicitly configure the switch to work with IGMPv2 or IGMPv3. When the IGMP version is not configured then the switch will work with both versions of IGMP.

Table 28-1: Configuring IGMP Snooping on Switch 1 and IGMP Querier on Switch 2

Configure IGMP Snooping (Switch 1)		

1.	`awplus#`
	`configure terminal`
		Enter Global Configuration mode.

Table 28-1: Configuring IGMP Snooping on Switch 1 and IGMP Querier on Switch 2

2.	`awplus(config)#`
	`ip igmp snooping`
		IGMP Snooping is enabled by default. Use this command only if you have previously disabled it.

3.	`awplus(config)#`
	`interface vlan1`
		Enter Interface Configuration mode for VLAN 1.

4.	`awplus(config-if)#`
	`ip igmp snooping mrouter interface port1.0.12`
		Configure port1.0.12 as a multicast router port to the IGMP Querier.

5.	`awplus(config-if)#`
	`exit`
		Return to Global Configuration mode.

Validate the configuration		
6.	`awplus#` `exit`	Return to Privileged Exec mode.

7.	`awplus#`
	`show ip igmp interface vlan1`
		Display the state of IGMP Snooping for VLAN 1.

8.	`awplus#`
	`show ip igmp groups`
		Display the multicast groups with receivers directly connected to the router.

9.	`awplus#`
	`show ip igmp snooping mrouter interface vlan1`
		Display the multicast router ports, both static and dynamic, in VLAN 1.

Configure IGMP Querier (Switch 2)		

1.	`awplus#`
	`configure terminal`
		Enter Global Configuration mode.

2.	`awplus(config)#`
	`interface vlan1`
		Enter Interface Configuration mode for VLAN 1.

3.	`awplus(config-if)#`
	`ip igmp`
		Enable IGMP on VLAN 1 and configure the switch as an IGMP Querier.

Validate the configuration		
4.	`awplus#` `exit`	Return to Privileged Exec mode.

5.	`awplus#`
	`show ip igmp interface vlan1`
		Display the state of IGMP Querier for VLAN 1.

6.	`awplus#`
	`show running-config`
		Display the current dynamic configuration of Switch 2.

Query Solicitation

Query Solicitation minimizes the loss of multicast data after a topology change on networks that use EPSR or spanning tree (STP, RSTP, or MSTP) for loop protection. Without Query Solicitation, when the underlying link layer topology changes, multicast data flow can stop for up to several minutes, depending on which port goes down and how much of the IGMP query interval remained at the time of the topology change. Query Solicitation greatly reduces this disruption.

Query Solicitation operates without configuration in AlliedWare Plus^{TM} switches running STP, RSTP, MSTP or EPSR. However, you may find it useful to manually enable Query Solicitation in loop-free networks running IGMP (see Speeding up IGMP convergence in a non-looped topology) and networks where not all switches support Query Solicitation (see Enabling Query Solicitation on multiple switches in a looped topology).

How Query Solicitation Works

Query Solicitation monitors STP, RSTP, MSTP and EPSR messages for topology changes. When it detects a change, it generates a special IGMP Leave message called a Query Solicit. The switch floods the Query Solicit message to all ports in every VLAN that Query Solicitation is enabled on. When the Querier receives the Query Solicit message, it sends out a General Query and waits for clients to respond with Membership Reports. These Reports update the snooping information throughout the network.

Query Solicit messages have a group address of 0.0.0.0.

Query Solicitation works by default (without you enabling it) on all VLANs on the root bridge in an STP instance and on all data VLANs on the master node in an EPSR instance. By default, the root bridge or master node always sends a Query Solicit message when any of the following events occur:

- an STP BPDU packet with the Topology Change (TC) flag arrives at the root bridge
- an STP port on a switch goes from a Discarding to Forwarding state
- the FDB gets flushed by EPSR

If necessary, you can make clients respond more quickly to the General Query by tuning the IGMP timers, especially the maximum response time advertised in IGMP queries using the ip igmp query-max-response-time command.

Query Solicitation Operation

When IGMP Snooping is enabled and EPSR or Spanning Tree changes the underlying link layer topology, this can interrupt multicast data flow for a significant length of time. This is because there is no way for switches in a network with interested clients to know where the traffic is available, due to the change in network topology. This change in network topology may take up to two IGMP Query intervals from the IGMP Querier, until the switches will know where to forward membership reports received by client hosts. During this time, those hosts will not receive multicast traffic.

Query solicitation prevents this by monitoring for any topology changes. When it detects a change, it generates a special IGMP Leave message known as a Query Solicit, and floods the Query Solicit message to all ports in every VLAN that query solicitation is enabled on. When the IGMP Querier receives the message, it responds by sending a General Query, which all IGMP listeners respond to. This refreshes snooped group membership information in the network.

Query solicitation reduces downtime to a negligible amount by triggering on topology changes. The generation of query solicitation messages in the network causes the IGMP Querier to send an IGMP Query immediately following a topology change resulting in the switches knowing where to look for the traffic and thus sending reports to the correct switch upstream, and thus allow the multicast data traffic to be recovered instantly.

Query solicitation functions by default (without you enabling it) on all VLANs on the root bridge in an STP instance and on all data VLANs on the master node in an EPSR instance. By default, the root bridge or master node always sends a Query Solicit message when the topology changes.

If you have multiple STP or EPSR instances, query solicitation only sends Query Solicit messages on VLANs in the instance that experienced a topology change.

In switches other than the STP root bridge or EPSR master node, query solicitation is disabled by default, but you can enable it by using the ip igmp snooping tcn query solicit command.

If you enable query solicitation on a switch other than the STP root bridge or EPSR master node, both that switch and the root or master send a Query Solicit message.

Once the Querier receives the Query Solicit message, it sends out a General Query and waits for responses, which update the snooping information throughout the network.

The ip igmp query-holdtime command can be configured on the IGMP Querier. This command introduces a brief delay between when the IGMP Querier receives the query solicit, and when it sends out the general query. Although this slightly reduces the speed with which the network recovers from the topology change, its does guard against a DoS (Denial of Service) attack. Without this delay, a malign host sending a stream of query solicits could cause the IGMP Querier to flood the network with IGMP Queries.

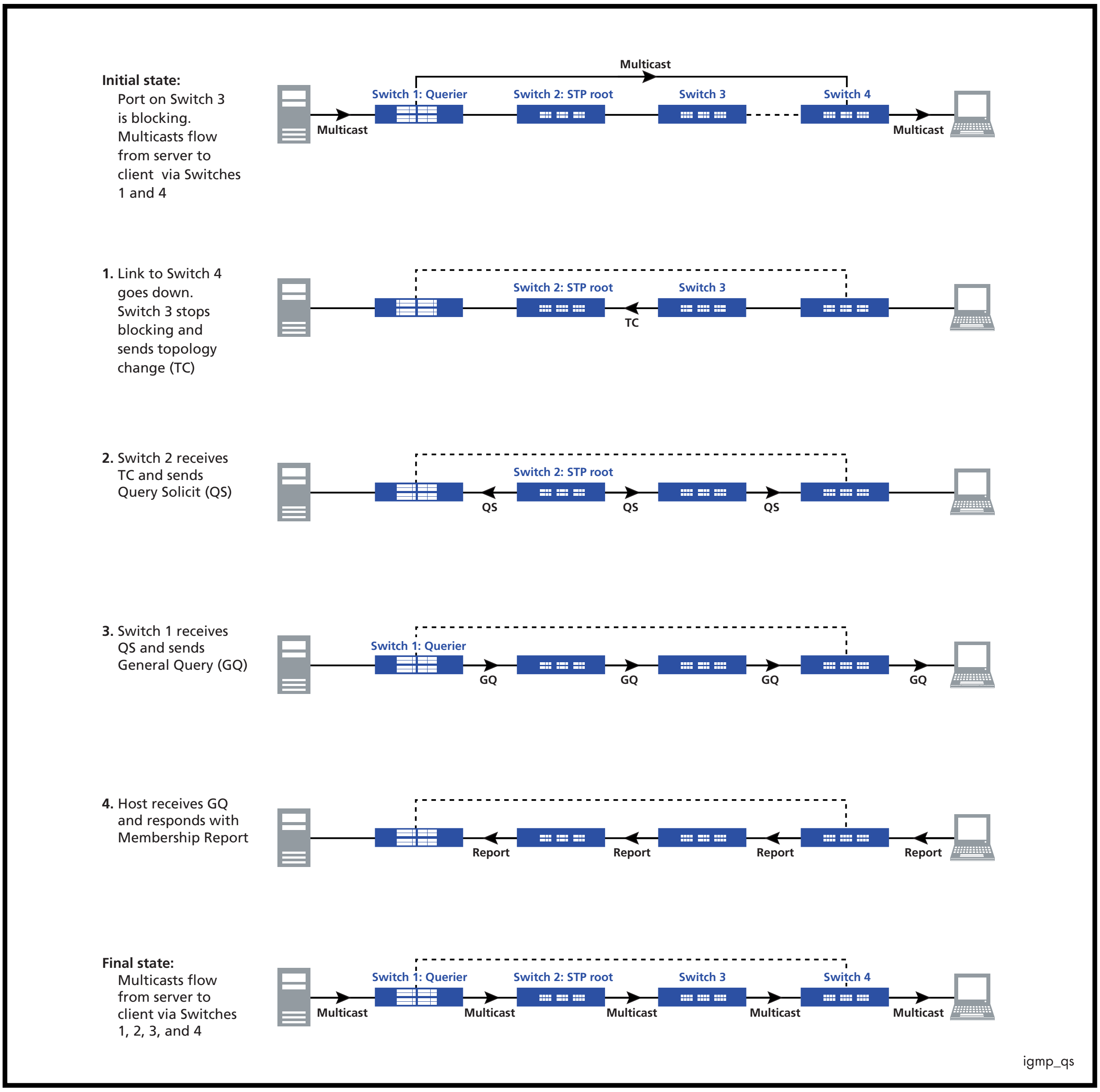
To get the network to converge faster, use the ip igmp query-max-response-time command and set a low response time value, such as one or two seconds, so that the clients will respond immediately with a report as a response to the IGMP Queries

On switches other than the STP root bridge or the EPSR master node, you can disable query solicitation by using the no variant of the ip igmp snooping tcn query solicit command. In addition, on all switches, you can disable query solicitation on a per-vlan basis using the no variant of the ip igmp snooping tcn query solicit command in Interface Configuration mode, after specifying a VLAN first in Interface Configuration mode.

To see whether query solicitation is on or off, check the Query Solicitation field in output of the show ip igmp interface command. You can view running and startup configurations with show running-config and show startup-config commands to see if Query Solicitation is enabled.

The following figure shows how Query Solicitation works when a port goes down.

Figure 28-3: Query Solicitation when a port goes down



Initial state:
Port on Switch 3 is blocking. Multicasts flow from server to client via Switches 1 and 4

1. Link to Switch 4 goes down. Switch 3 stops blocking and sends topology change (TC)

2. Switch 2 receives TC and sends Query Solicit (QS)

3. Switch 1 receives QS and sends General Query (GQ)

4. Host receives GQ and responds with Membership Report

Final state:
Multicasts flow from server to client via Switches 1, 2, 3, and 4

Speeding up IGMP convergence in a non-looped topology

For loop-free networks running IGMP, where it may take up to two minutes for multicasting to recover in a non-looped topology after a port comes back up, you can speed up convergence by enabling RSTP using the **spanning-tree mode** and **spanning-tree enable** commands.

RSTP enables the network to use Query Solicitation by default, and means that multicasting should resume within seconds, not minutes, of the link coming up.

Enabling Query Solicitation on multiple switches in a looped topology

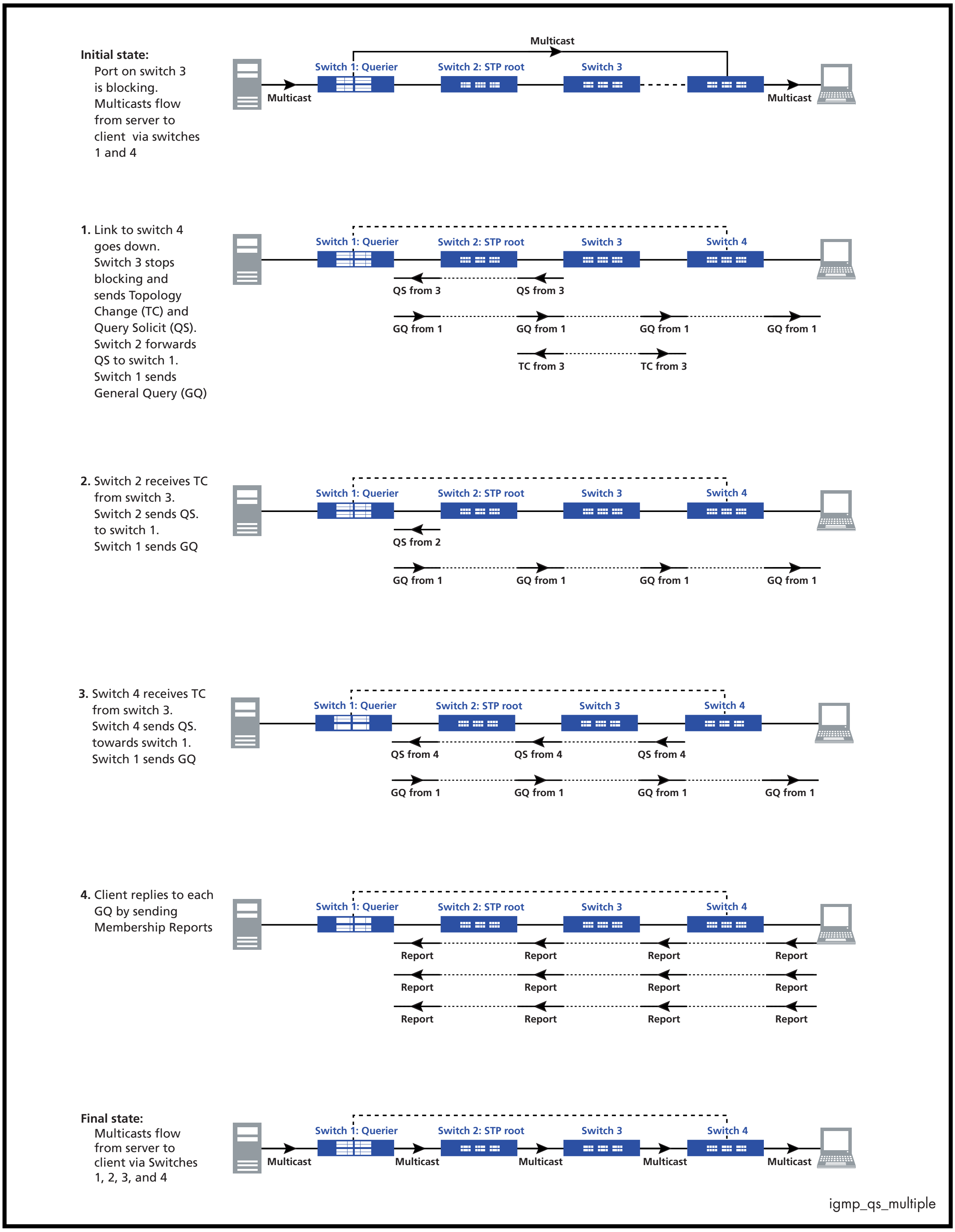
On networks that use spanning tree or EPSR, Query Solicitation is not normally required on switches other than the STP root bridge or EPSR master node. Therefore, it is only enabled by default on the root bridge and the master node.

However, in some networks you may need to turn on Query Solicitation on all switches - for example, if the network includes other switches that do not support Query Solicitation and therefore the STP root bridge may be a switch that does not send Query Solicit messages. To enable Query Solicitation, use the **ip igmp snooping tcn query solicit** command.

Every switch that has Query Solicitation enabled sends a Query Solicit message when it detects a topology change. Enabling it on multiple switches means you get multiple messages, but has no other disadvantage.

The following figure shows the packet flow for a four-switch network with Query Solicitation enabled on all the switches.

Figure 28-4: Packet flow for a four switch network with Query Solicitation enabled



Initial state:
Port on switch 3 is blocking. Multicasts flow from server to client via switches 1 and 4

1. Link to switch 4 goes down. Switch 3 stops blocking and sends Topology Change (TC) and Query Solicit (QS). Switch 2 forwards QS to switch 1. Switch 1 sends General Query (GQ)

2. Switch 2 receives TC from switch 3. Switch 2 sends QS. to switch 1. Switch 1 sends GQ

3. Switch 4 receives TC from switch 3. Switch 4 sends QS. towards switch 1. Switch 1 sends GQ

4. Client replies to each GQ by sending Membership Reports

Final state:
Multicasts flow from server to client via Switches 1, 2, 3, and 4

Chapter 29: IGMP and IGMP Snooping Commands

Introduction

The Internet Group Management Protocol (IGMP) module includes the IGMP Proxy service and IGMP Snooping functionality. Some of the following commands may have commonalities and restrictions. These are described under the Usage section for each command.

Command List

This chapter provides an alphabetical reference of configure, clear, and show commands related to Internet Group Management Protocol (IGMP).

clear ip igmp

Use this command to clear all IGMP group membership records on all VLAN interfaces.

Syntax

```
clear ip igmp
```

Mode Privileged Exec

Usage This command applies to VLAN interfaces configured for IGMP, IGMP Snooping, or IGMP Proxy.

Example

```
awplus# clear ip igmp
```

Validation Commands

show ip igmp interface
show running-config

Related Commands

clear ip igmp group
clear ip igmp interface

clear ip igmp group

Use this command to clear IGMP group membership records for a specific group on either all VLAN interfaces, a single VLAN interface, or for a range of VLAN interfaces.

Syntax

```
clear ip igmp group *
clear ip igmp group <ip-address> <interface>
```

Parameter	Description
*	Clears all groups on all VLAN interfaces. This is an alias to the **clear ip igmp** command.
<ip-address>	Specifies the group whose membership records will be cleared from all VLAN interfaces, entered in the form A.B.C.D.
<interface>	Specifies the name of the VLAN interface; all groups learned on this VLAN interface are deleted.

Mode Privileged Exec

Usage This command applies to groups learned by IGMP, IGMP Snooping, or IGMP Proxy.

In addition to the group a VLAN interface can be specified. Specifying this will mean that only entries with the group learnt on the interface will be deleted.

Examples

```
awplus# clear ip igmp group *
```

```
awplus# clear ip igmp group 224.1.1.1 vlan1
```

Validation Commands

show ip igmp interface
show running-config

Related Commands

clear ip igmp
clear ip igmp interface

clear ip igmp interface

Use this command to clear IGMP group membership records on a particular VLAN interface.

Syntax

```
clear ip igmp interface <interface>
```

Parameter	Description
<interface>	Specifies the name of the VLAN interface. All groups learned on this VLAN interface are deleted.

Mode Privileged Exec

Usage This command applies to interfaces configured for IGMP, IGMP Snooping, or IGMP Proxy.

Example

```
awplus# clear ip igmp interface vlan1
```

Validation Commands

show ip igmp interface
show running-config

Related Commands

clear ip igmp
clear ip igmp group

debug igmp

Use this command to enable debugging of either all IGMP or a specific component of IGMP.

Use the **no** variant of this command to disable all IGMP debugging, or debugging of a specific component of IGMP.

Syntax

```
debug igmp {all|decode|encode|events|fsm|tib}
```

```
no debug igmp {all|decode|encode|events|fsm|tib}
```

Parameter	Description
`all`	Enable or disable all debug options for IGMP
`decode`	Debug of IGMP packets that have been received
`encode`	Debug of IGMP packets that have been sent
`events`	Debug IGMP events
`fsm`	Debug IGMP Finite State Machine (FSM)
`tib`	Debug IGMP Tree Information Base (TIB)

Modes Privileged Exec and Global Configuration

Usage This command applies to interfaces configured for IGMP, IGMP Snooping, or IGMP Proxy.

Example

```
        awplus# configure terminal
awplus(config)# debug igmp all
```

Related Commands show debugging igmp
undebug igmp

ip igmp

Use this command to enable IGMP on an interface. The command configures the device as an IGMP querier.

Use the **no** variant of this command to return all IGMP related configuration to the default on this interface.

Syntax

```
ip igmp
```

```
no ip igmp
```

Default Disabled

Mode Interface Configuration for a VLAN interface.

Usage This command can only be configured on VLAN interfaces, and will have no effect on IGMP Proxy or IGMP Snooping configuration.

Example

```
          awplus# configure terminal
  awplus(config)# interface vlan1
awplus(config-if)# ip igmp
```

Validation Commands

show ip igmp interface
show running-config

ip igmp access-group

This command adds an access control list to a VLAN interface configured for IGMP, IGMP Snooping, or IGMP Proxy. The access control list is used to control and filter the multicast groups learnt on the VLAN interface.

The **no** variant of this command disables the access control filtering on the interface.

Syntax

```
ip igmp access-group {<access-list-number>|<access-list-name>}
```

```
no ip igmp access-group
```

Parameter	Description
<access-list-number>	Standard IP access-list number, in the range <1-99>.
<access-list-name>	Standard IP access-list name.

Default By default there are no access lists configured on any interface.

Mode Interface Configuration for a VLAN interface.

Usage This command applies to VLAN interfaces configured for IGMP, IGMP Snooping, or IGMP Proxy.

Example In the following example, hosts serviced by VLAN 1 can only join the group 225.2.2.2:

```
        awplus# configure terminal
  awplus(config)# access-list 1 permit 225.2.2.2 0.0.0.0
  awplus(config)# interface vlan1
awplus(config-if)# ip igmp access-group 1
```

ip igmp immediate-leave

In IGMP version 2, use this command to minimize the leave latency of IGMP memberships for specified multicast groups. The specified access list number or name defines the multicast groups in which the immediate leave feature is enabled.

Use the **no** variant of this command to disable this feature.

Syntax

```
ip igmp immediate-leave group-list {<access-list-number>|<access-
    list-number-expanded>|<access-list-name>}

no ip igmp immediate-leave
```

Parameter	Description
<access-list-number>	Access-list number, in the range <1-99>.
<access-list-number-expanded>	Access-list number (expanded range), in the range <1300-1999>.
<access-list-name>	Standard IP access-list name.

Default Disabled by default.

Mode Interface Configuration for a VLAN interface.

Usage This command applies to interfaces configured for IGMP, IGMP Snooping, or IGMP Proxy.

Example The following example shows how to enable the immediate-leave feature on an interface for a specific range of multicast groups

```
          awplus# configure terminal
  awplus(config)# interface vlan1
awplus(config-if)# ip igmp immediate-leave group-list 34
awplus(config-if)# exit
  awplus(config)# access-list 34 permit 225.192.20.0 0.0.0.255
```

Related Commands ip igmp last-member-query-interval

ip igmp last-member-query-count

Use this command to set the last-member query-count value for an interface.

Use the **no** variant of this command to return to the default on an interface.

Syntax

```
ip igmp last-member-query-count <2-7>

no ip igmp last-member-query-count
```

Parameter	Description
<2-7>	Last member query count value.

Default The default last member query count value is 2.

Mode Interface Configuration for a VLAN interface.

Usage This command applies to interfaces configured for IGMP, IGMP Snooping, or IGMP Proxy.

Example

```
          awplus# configure terminal
  awplus(config)# interface vlan1
awplus(config-if)# ip igmp last-member-query-count 3
```

Validation Commands

show ip igmp interface
show running-config

Related Commands

ip igmp last-member-query-interval
ip igmp startup-query-count

ip igmp last-member-query-interval

Use this command to configure the frequency at which the router sends IGMP group specific host query messages.

Use the **no** variant of this command to set this frequency to the default.

Syntax

```
ip igmp last-member-query-interval <interval>

no ip igmp last-member-query-interval
```

Parameter	Description
<interval>	The frequency in milliseconds, in the range <1000-25500>, at which IGMP group-specific host query messages are sent.

Default 1000 milliseconds

Mode Interface Configuration for a VLAN interface.

Usage This command applies to interfaces configured for IGMP, IGMP Snooping, or IGMP Proxy.

Example The following example changes the IGMP group-specific host query message interval to 2 seconds (2000 milliseconds):

```
          awplus# configure terminal
  awplus(config)# interface vlan1
awplus(config-if)# ip igmp last-member-query-interval 2000
```

Validation Commands

show ip igmp interface
show running-config

Related Commands

ip igmp immediate-leave
ip igmp last-member-query-count

ip igmp limit

Use this command to configure the limit on the maximum number of group membership entries for the device as a whole or for the specified interface (if in interface mode). Once the specified number of group memberships is reached, all further membership reports will be ignored. Optionally, you can configure an access-list to stop certain address(es) from being subject to the limit.

The limit is dependent on the MTU (Maximum Transmission Unit) of the interface, which is the size in bytes of the largest packet that a network protocol can transmit. Typically for an ethernet channel with an MTU of 1500 the igmp group membership limit will be 183 groups, because each igmp group membership is 8 bytes.

Use the **no** variant of this command to unset the limit and any specified exception access-list.

Syntax

```
ip igmp limit <limitvalue> [except {<access-list-number>|<access-
    list-number-expanded>|<access-list-name>}]
```

```
no ip igmp limit
```

Parameter	Description
<limitvalue>	<2-512> Maximum number of group membership entries.
<access-list-number>	Access-list number, in the range <1-99>.
<access-list-number-expanded>	Access-list number (expanded range), in the range <1300-1999>.
<access-list-name>	Standard IP access-list name.

Default The default limit, which is reset by the **no** variant of this command, is the same as maximum number of group membership entries that can be learned with the **ip igmp limit** command.

The default limit of group membership entries that can be learned is 512 entries.

Mode Global Configuration and Interface Configuration for a VLAN interface.

Usage This command applies to interfaces configured for IGMP, IGMP Snooping, or IGMP Proxy.

Examples The following example configures an IGMP limit of `100` group membership entries across all interfaces on which IGMP is enabled, and excludes group `224.1.1.1` from this limitation:

```
         awplus# configure terminal
 awplus(config)# access-list 1 permit 224.1.1.1 0.0.0.0
 awplus(config)# ip igmp limit 100 except 1
```

The following example configures an IGMP limit of 100 group membership entries on vlan1:

```
          awplus# configure terminal
  awplus(config)# interface vlan1
awplus(config-if)# ip igmp limit 100
```

ip igmp mroute-proxy

Use this command to enable IGMP mroute proxy on this downstream interface and associate it with the upstream proxy service interface.

Use the **no** variant of this command to remove the association with the proxy-service interface.

Syntax

```
ip igmp mroute-proxy <interface>
```

```
no ip igmp mroute-proxy
```

Parameter	Description
<interface>	The name of the VLAN interface.

Mode Interface Configuration for a VLAN interface.

Usage You must also enable the IGMP proxy service on the upstream interface, using the ip igmp proxy-service command. You can associate one or more downstream mroute proxy interfaces on the device with a single upstream proxy service interface. This downstream mroute proxy interface listens for IGMP reports, and forwards them to the upstream IGMP proxy service interface.

Example The following example configures the `vlan1` interface as the upstream proxy-service interface for the downstream interface, `vlan2`.

```
           awplus# configure terminal
   awplus(config)# interface vlan2
awplus(config-if)# ip igmp mroute-proxy vlan1
```

Related Commands ip igmp proxy-service

ip igmp proxy-service

Use this command to enable the VLAN interface to be the upstream IGMP proxy-service interface for the device. All associated downstream IGMP mroute proxy interfaces on this device will have their memberships consolidated on this proxy service interface, according to IGMP host-side functionality.

Use the **no** variant of this command to remove the designation of the VLAN interface as an upstream proxy-service interface.

Syntax

```
ip igmp proxy-service
```

```
no ip igmp proxy-service
```

Mode Interface Configuration for a VLAN interface.

Usage This command is used with the ip igmp mroute-proxy command to enable forwarding of IGMP reports to a proxy service interface for all forwarding entries for this interface. You must also enable the downstream IGMP mroute proxy interfaces on this device using the command ip igmp mroute-proxy.

Example The following example designates the `vlan1` interface as the upstream proxy-service interface.

```
          awplus#  configure terminal
  awplus(config)#  interface vlan1
awplus(config-if)#  ip igmp proxy-service
```

Related Commands ip igmp mroute-proxy

ip igmp querier-timeout

Use this command to configure the timeout period before the device takes over as the querier for the VLAN interface after the previous querier has stopped querying.

Use the **no** variant of this command to restore the default.

Syntax

```
ip igmp querier-timeout <timeout>

no ip igmp querier-timeout
```

Parameter	Description
<timeout>	IGMP querier timeout interval value in seconds, in the range <1-65535>.

Default The default timeout interval is 255 seconds.

Mode Interface Configuration for a VLAN interface.

Usage This command applies to VLAN interfaces configured for IGMP. The timeout value should not be less than the current active querier's general query interval.

Example The following example configures the device to wait 130 seconds from the time it received the last query before it takes over as the querier for the interface:

```
          awplus# configure terminal
  awplus(config)# interface vlan20
awplus(config-if)# ip igmp querier-timeout 130
```

Validation Commands

show ip igmp interface
show running-config

Related Commands ip igmp query-interval

ip igmp query-holdtime

This command sets the time that an IGMP Querier waits after receiving a query solicitation before it sends an IGMP Query. IGMP General Query messages will not be sent during the hold time interval.

Use the **no** variant of this command to return to the default query hold time period.

Syntax

```
ip igmp query-holdtime <interval>
```

```
no ip igmp query-holdtime
```

Parameter	Description
`<interval>`	Query interval value in milliseconds, in the range <100-5000>.

Default By default the delay before sending IGMP General Query messages is 500 milliseconds.

Mode Interface Configuration for a VLAN interface.

Usage Use this command to configure a value for the IGMP query hold time in the current network. IGMP Queries can be generated after receiving Query Solicitation (QS) packets and there is a possibility of a DoS (Denial of Service) attack if a stream of Query Solicitation (QS) packets are sent to the IGMP Querier, eliciting a rapid stream of IGMP Queries. This command applies to interfaces on which the switch is acting as an IGMP Querier.

Use the ip igmp query-interval command when a delay for IGMP general query messages is required and IGMP general query messages are required. The **ip igmp query-holdtime** command stops IGMP query messages during the configured holdtime interval, so the rate of IGMP Queries that can be sent out of an interface can be restricted.

See “Query Solicitation” on page 28.7 for introductory information about the Query Solicitation feature.

Note This command will function on your switch in the stand-alone mode. but is not supported when the switch forms part of a VCS Stack.

Examples To set the IGMP query holdtime to `900` ms for `vlan20`, use the following commands:

```
          awplus#  configure terminal
  awplus(config)#  interface vlan20
awplus(config-if)# ip igmp query-holdtime 900
```

To reset the IGMP query holdtime to the default (500 ms) for `vlan10`, use the following commands:

```
          awplus#  configure terminal
  awplus(config)#  interface vlan10
awplus(config-if)# no ip igmp query-holdtime
```

Validation Commands	show ip igmp interface show running-config
Related Commands	ip igmp query-interval ip igmp snooping tcn query solicit

ip igmp query-interval

Use this command to configure the period for sending IGMP General Query messages. The IGMP query interval specifies the time between IGMP General Query messages being sent.

Use the **no** variant of this command to return to the default query interval period.

Note The IGMP query interval must be greater than IGMP query maximum response time.

Syntax

```
ip igmp query-interval <interval>

no ip igmp query-interval
```

Parameter	Description
`<interval>`	Query interval value in seconds, in the range `<2-18000>`.

Default The default IGMP query interval is 125 seconds.

Mode Interface Configuration for a VLAN interface.

Usage This command applies to interfaces configured for IGMP. Note that the IGMP query interval is automatically set to a greater value than the IGMP query max response time.

For example, if you set the IGMP query max response time to 2 seconds using the ip igmp query-max-response-time command, and the IGMP query interval is currently less than 3 seconds, then the IGMP query interval period will be automatically reconfigured to be 3 seconds, so it is greater than the IGMP query maximum response time.

Use the **ip igmp query-interval** command when a non-default interval for IGMP General Query messages is required.

The ip igmp query-holdtime command can occasionally delay the sending of IGMP Queries.

Examples The following example changes the period between IGMP host-query messages to 3 minutes (180 seconds):

```
          awplus# configure terminal
  awplus(config)# interface vlan20
awplus(config-if)# ip igmp query-interval 180
```

The following example resets the period between sending IGMP host-query messages to the default (125 seconds):

```
          awplus# configure terminal
  awplus(config)# interface vlan20
awplus(config-if)# no ip igmp query-interval
```

Validation Commands

show ip igmp interface
show running-config

Related Commands

ip igmp query-holdtime
ip igmp query-max-response-time
ip igmp startup-query-interval

ip igmp query-max-response-time

Use this command to configure the maximum response time advertised in IGMP Queries.

Use the **no** variant of this command to restore the default.

Note The IGMP query maximum response time must be less than the IGMP query interval.

Syntax

```
ip igmp query-max-response-time <response-time>

no ip igmp query-max-response-time
```

Parameter	Description
<response-time>	Response time value in seconds, in the range <1-3180>.

Default The default IGMP query maximum response time is 10 seconds.

Mode Interface Configuration for a VLAN interface.

Usage This command applies to interfaces configured for IGMP. Note that the IGMP query interval is automatically set to a greater value than the IGMP query maximum response time.

For example, if you set the IGMP query interval to 3 seconds using the ip igmp query-interval command, and the current IGMP query interval is less than 3 seconds, then the IGMP query maximum response time will be automatically reconfigured to be 2 seconds, so it is less than the IGMP query interval time.

To get the network to converge faster, use the **ip igmp query-max-response-time** command and set a low response time value, such as one or two seconds, so that the clients will respond immediately with a report as a response to the IGMP Queries

Examples The following example configures a maximum response time of 8 seconds:

```
           awplus#  configure terminal
   awplus(config)#  interface vlan1
awplus(config-if)#  ip igmp query-max-response-time 8
```

The following example restores the default maximum response time of 10 seconds:

```
           awplus#  configure terminal
   awplus(config)#  interface vlan1
awplus(config-if)#  no ip igmp query-max-response-time
```

Validation Commands show ip igmp interface
show running-config

Related Commands ip igmp query-interval

ip igmp ra-option (Router Alert)

Use this command to enable strict Router Alert (RA) option validation. With strict RA option enabled, IGMP packets without RA options are ignored.

Use the **no** variant of this command to disable strict RA option validation.

Syntax

```
ip igmp ra-option
no ip igmp ra-option
```

Default The default state of RA validation is unset.

Mode Interface Configuration for a VLAN interface.

Usage This command applies to interfaces configured for IGMP and IGMP Snooping.

Example

```
        awplus# configure terminal
awplus(config)# interface vlan20
awplus(config-if)# ip igmp ra-option
```

ip igmp robustness-variable

Use this command to change the robustness variable value on a VLAN interface.

Use the **no** variant of this command to return to the default on an interface.

Syntax

```
ip igmp robustness-variable <1-7>

no ip igmp robustness-variable
```

Parameter	Description
<*1-7*>	The robustness variable value.

Default The default robustness variable value is 2.

Mode Interface Configuration for a VLAN interface.

Usage This command applies to interfaces configured for IGMP and IGMP Snooping.

Examples

```
           awplus# configure terminal
   awplus(config)# interface vlan20
awplus(config-if)# ip igmp robustness-variable 3
```

```
           awplus# configure terminal
   awplus(config)# interface vlan20
awplus(config-if)# no ip igmp robustness-variable
```

Validation Commands

show ip igmp interface
show running-config

ip igmp snooping

Use this command to enable IGMP Snooping. When this command is used in the Global Configuration mode, IGMP Snooping is enabled at the switch level. When this command is used in Interface Configuration mode, IGMP Snooping is enabled for the specified VLANs.

Use the **no** variant of this command to either globally disable IGMP Snooping, or disable IGMP Snooping on a specified interface.

Note IGMP snooping cannot be disabled on an interface if IGMP snooping has already been disabled globally. IGMP snooping can be disabled on both an interface and globally if disabled on the interface first and then disabled globally.

Syntax

```
ip igmp snooping

no ip igmp snooping
```

Default

By default, IGMP Snooping is enabled both globally and on all VLANs.

Mode

Global Configuration and Interface Configuration for a VLAN interface.

Usage

For IGMP snooping to operate on particular VLAN interfaces, it must be enabled both globally by using this command in Global Configuration mode, and on individual VLAN interfaces by using this command in Interface Configuration mode (both are enabled by default.)

Examples

```
        awplus# configure terminal
awplus(config)# ip igmp snooping
awplus(config)# interface vlan1
awplus(config-if)# ip igmp snooping
```

Related Commands

show ip igmp interface
show running-config

ip igmp snooping fast-leave

Use this command to enable IGMP Snooping fast-leave processing. Fast-leave processing is analogous to immediate-leave processing. The IGMP group-membership entry is removed as soon as an IGMP leave group message is received, without sending out a group-specific query.

Use the **no** variant of this command to disable fast-leave processing.

Syntax

```
ip igmp snooping fast-leave
```

```
no ip igmp snooping fast-leave
```

Default IGMP Snooping fast-leave processing is disabled.

Mode Interface Configuration for a VLAN interface.

Usage This IGMP Snooping command can only be configured on VLAN interfaces.

Example This example shows how to enable fast-leave processing on a VLAN.

```
          awplus# configure terminal
  awplus(config)# interface vlan1
awplus(config-if)# ip igmp snooping fast-leave
```

Validation Commands

show ip igmp interface
show running-config

ip igmp snooping mrouter

Use this command to statically configure the specified port in the VLAN as a multicast router port for IGMP Snooping in that VLAN. This command applies to interfaces configured for IGMP Snooping.

Use the **no** variant of this command to remove the static configuration of the port as a multicast router port.

Syntax

```
ip igmp snooping mrouter interface <port>

no ip igmp snooping mrouter interface <port>
```

Parameter	Description
<port>	The port may be a switch port (e.g. `port1.0.4`), a static channel group (e.g. `sa3`), or a dynamic (LACP) channel group (e.g. `po4`).

Mode Interface Configuration for a VLAN interface.

Usage This IGMP Snooping command can only be configured on VLAN interfaces.

Example

```
          awplus# configure terminal
  awplus(config)# interface vlan1
awplus(config-if)# ip igmp snooping mrouter interface port1.0.2
```

Related Commands show ip igmp snooping mrouter

ip igmp snooping querier

Use this command to enable IGMP querier operation on a VLAN when no multicast routing protocol is configured in the VLAN. When enabled, the IGMP Snooping querier sends out periodic IGMP queries for all interfaces on that VLAN. This command applies to interfaces configured for IGMP Snooping.

Use the **no** variant of this command to disable IGMP querier configuration.

Syntax

```
ip igmp snooping querier
```

```
no ip igmp snooping querier
```

Mode Interface Configuration for a VLAN interface.

Usage This command can only be configured on VLAN interfaces.

The IGMP Snooping querier uses the `0.0.0.0` Source IP address because it only masquerades as a proxy IGMP querier for faster network convergence.

It does not start, or automatically cease, the IGMP Querier operation if it detects query message(s) from a multicast router.

If an IP address is assigned to a VLAN, which has IGMP querier enabled on it, then the IGMP Snooping querier uses the VLAN's IP address as the Source IP Address in IGMP queries.

The IGMP Snooping Querier will not stop sending IGMP Queries if there is another IGMP Snooping Querier in the network with a lower Source IP Address.

Note Do not enable the IGMP Snooping Querier feature on a Layer 2 switch when there is an operational IGMP Querier in the network.

Example

```
          awplus#  configure terminal
  awplus(config)#  interface vlan1
awplus(config-if)#  ip igmp snooping querier
```

Validation Commands show ip igmp interface
show running-config

ip igmp snooping report-suppression

Use this command to enable report suppression for IGMP versions 1 and 2. This command applies to interfaces configured for IGMP Snooping.

Report suppression stops reports being sent to an upstream multicast router port when there are already downstream ports for this group on this interface.

Use the **no** variant of this command to disable report suppression.

Syntax

```
ip igmp snooping report-suppression

no ip igmp snooping report-suppression
```

Default Report suppression does not apply to IGMPv3, and is turned on by default for IGMPv1 and IGMPv2 reports.

Mode Interface Configuration for a VLAN interface.

Usage This command can only be configured on VLAN interfaces.

Example This example shows how to enable report suppression for IGMPv2 reports.

```
          awplus# configure terminal
  awplus(config)# interface vlan1
awplus(config-if)# ip igmp version 2
awplus(config-if)# ip igmp snooping report-suppression
```

Validation Commands

show ip igmp interface
show running-config

ip igmp snooping routermode

Use this command to set the destination IP addresses as a router multicast address, according to the routermode (all multicast addresses, default multicast addresses, specified multicast addresses).

Use the **no** variant of this command to the default. You can also remove a specified IP address from a custom list of multicast addresses.

Syntax

```
ip igmp snooping routermode {all|default|ip|multicastrouter|address
    <ip-address>}
```

```
no ip igmp snooping routermode [address <ip-address>]
```

Parameter	Description
`all`	All reserved multicast addresses (224.0.0.x). Packets from all possible addresses in range 224.0.0.x are set as routers.
`default`	Default set of reserved multicast addresses. Packets from 224.0.0.1, 224.0.0.2, 224.0.0.4, 224.0.0.5, 224.0.0.6, 224.0.0.9, 224.0.0.13, 224.0.0.15 and 224.0.0.24 are set as routers.
`ip`	Custom reserved multicast addresses. Custom IP address in the 224.0.0.x range are set as router multicast addresses using the **ip igmp snooping routermode address** command.
`address`	Specify the multicast address in the 224.0.0.x range for use after issuing an **ip igmp snooping routermode ip** command
`<ip-address>`	IPv4 multicast address (224.0.0.x)

Default The default routermode is **default** not **all** and shows the below reserved multicast addresses:

```
Router mode.............Def
Reserved multicast address
         224.0.0.1
         224.0.0.2
         224.0.0.4
         224.0.0.5
         224.0.0.6
         224.0.0.9
         224.0.0.13
         224.0.0.15
          224.0.0.24
```

Mode Global Configuration

Examples To set **ip igmp snooping routermode** for all default reserved addresses enter:.

```
awplus(config)# ip igmp snooping routermode default
```

To remove the multicast address 224.0.0.5 from the custom list of multicast addresses enter:.

```
awplus(config)# no ip igmp snooping routermode address
                224.0.0.5
```

Related commands show ip igmp snooping routermode

ip igmp snooping tcn query solicit

Use this command to enable IGMP (Internet Group Management Protocol) Snooping TCN (Topology Change Notification) Query Solicitation feature. When this command is used in the Global Configuration mode, Query Solicitation is enabled for the specified VLANs.

Use the **no** variant of this command to disable IGMP Snooping TCN Query Solicitation. When the no variant of this command is used in Interface Configuration mode, this overrides the Global Configuration mode setting and Query Solicitation is disabled for the specified VLANs.

Syntax

```
ip igmp snooping tcn query solicit

no ip igmp snooping tcn query solicit
```

Default IGMP Snooping TCN Query Solicitation is disabled by default on the switch, unless the switch is the Master Node in an EPSR ring, or is the Root Bridge in a Spanning Tree.

When the switch is the Master Node in an EPSR ring, or the switch is the Root Bridge in a Spanning Tree, then IGMP Snooping TCN Query Solicitation is enabled by default and cannot be disabled using the Global Configuration mode command. However, Query Solicitation can be disabled for specified VLANs using this command from the Interface Configuration mode. Select the VLAN you want to disable in Interface Configuration mode then issue the no variant of this command to disable the specified VLAN without disabling this feature for other VLANs.

Mode Global Configuration and Interface Configuration for a VLAN interface.

Usage Once enabled, if the switch is not an IGMP Querier, on detecting a topology change, the switch generates IGMP Query Solicit messages that are sent to all the ports of the vlan configured for IGMP Snooping on the switch.

On a switch that is not the Master Node in an EPSR ring or the Root Bridge in a Spanning Tree, Query Solicitation can be disabled using the **no** variant of this command after being enabled.

If the switch that detects a topology change is an IGMP Querier then the switch will generate an IGMP Query message.

Note that the **no** variant of this command when issued in Global Configuration mode has no effect on a switch that is the Master Node in an EPSR ring or on a switch that is a Root Bridge in a Spanning Tree. Query Solicitation is not disabled for the switch these instances. However, Query Solicitation can be disabled on a per-vlan basis from the Interface Configuration mode.

See the below state table that shows when Query Solicit messages are sent in these instances:

Command issued from Global Configuration	Switch is STP Root Bridge or the EPSR Master Node	Command issued from Interface Configuration	IGMP Query Solicit message sent on VLAN
No	Yes	Yes	Yes
Yes	Yes	No	No
Yes	Yes	Yes	Yes

See "Query Solicitation" on page 28.7 for introductory information about the Query Solicitation feature.

Note This command will function on your switch in the stand-alone mode. but is not supported when the switch forms part of a VCS Stack.

Examples This example shows how to enable IGMP Snooping TCN Query Solicitation on a switch:

```
        awplus# configure terminal
awplus(config)# ip igmp snoopig tcn query solicit
```

This example shows how to disable IGMP Snooping TCN Query Solicitation on a switch:

```
        awplus# configure terminal
awplus(config)# no ip igmp snooping tcn query solicit
```

This example shows how to enable IGMP Snooping TCN Query Solicitation for interface **vlan2**:

```
           awplus# configure terminal
   awplus(config)# interface vlan2
awplus(config-if)# ip igmp snoopig tcn query solicit
```

This example shows how to disable IGMP Snooping TCN Query Solicitation for interface **vlan2**:

```
           awplus# configure terminal
   awplus(config)# interface vlan2
awplus(config-if)# no ip igmp snooping tcn query solicit
```

Validation Commands show ip igmp interface
show running-config

Related Commands ip igmp query-holdtime

ip igmp source-address-check

This command enables the checking of the Source Address for an IGMP Report, rejecting any IGMP Reports originating on devices outside of the local subnet.

Use the **no** variant of this command to disable the checking of the Source Address for an IGMP Report, which allows IGMP Reports from devices outside of the local subnet.

Syntax

```
ip igmp source-address-check
```

```
no ip igmp source-address-check
```

Default Source address checking for IGMP Reports is enabled by default.

Mode Interface Configuration for a VLAN interface.

Usage This is a security feature, and should be enabled unless IGMP Reports from outside the local subnet are expected, for example, if Multicast VLAN Registration is active in the network.

The **no** variant of this command is required to disable the IGMP Report source address checking feature in networks that use Multicast VLAN Registration to allow IGMP Reports from devices outside of the local subnet.

Examples To deny IGMP Reports from outside the current subnet, use the following commands:

```
          awplus#  configure terminal
  awplus(config)#  interface vlan2
awplus(config-if)# ip igmp source-address-check
```

To allow IGMP Reports from outside the current subnet, use the following commands:

```
          awplus#  configure terminal
  awplus(config)#  interface vlan2
awplus(config-if)# no ip igmp source-address-check
```

Validation Commands

show ip igmp interface
show running-config

ip igmp static-group

Use this command to statically configure multicast group membership entries on a VLAN interface, or to statically forward a multicast channel out a particular port or port range.

To statically add only a group membership, do not specify any parameters.

To statically add a (*,g) entry to forward a channel out of a port specify only the multicast group address and the switch port range.

To statically add a (s,g) entry to forward a channel out of a port specify the multicast group address, the source IP address, and the switch port range.

Use the **no** variant of this command to delete static group membership entries.

Syntax

```
ip igmp static-group <ip-address> [source {<ip-source-addr>}]
    [interface <port>]

no ip igmp static-group <ip-address> [source {<ip-source-addr>}]
    [interface <port>]
```

Parameter	Description
<ip-address>	Standard IP Multicast group address, entered in the form A.B.C.D, to be configured as a static group member.
`source`	Optional.
<ip-source-addr>	Standard IP source address, entered in the form A.B.C.D, to be configured as a static source from where multicast packets originate.
`interface`	Use this parameter to specify a specific switch port or switch port range to statically forward the multicast group out of. If not used, static configuration is applied on all ports in the VLAN.
<port>	The port or port range to statically forward the group out of. The port may be a switch port (e.g. `port1.0.4`), a static channel group (e.g. `sa3`), or a dynamic (LACP) channel group (e.g. `po4`).

Mode Interface Configuration for a VLAN interface.

Usage This command applies to IGMP operation on a specific interface to statically add group and/or source records; or to IGMP Snooping on a VLAN interface to statically add group and/or source records.

Example The following example show how to statically add group and source records for IGMP:

```
          awplus#  configure terminal
  awplus(config)#  interface vlan3
awplus(config-if)#  ip igmp static-group 226.1.2.4 source
                    10.2.3.4
```

ip igmp startup-query-count

Use this command to configure the IGMP startup query count for a VLAN interface in Interface Configuration mode. The IGMP startup query count is the number of IGMP General Query messages sent by a querier at startup. The default IGMP startup query count is 2.

Use the **no** variant of this command to remove the configured IGMP startup query count for a VLAN interface in Interface Configuration mode.

Syntax

```
ip igmp startup-query-count <startup-query-count>
```

```
no ip igmp startup-query-count
```

Parameter	Description
<startup-query-count>	Specify the IGMP startup query count for a VLAN interface in the range <2-10> where 2 is the default IGMP query count.

Default The default IGMP startup query count is 2.

Mode Interface Configuration for a VLAN interface.

Examples The following example shows how to configure the IGMP startup query count to 4 for VLAN interface vlan3:

```
          awplus# configure terminal
  awplus(config)# interface vlan3
awplus(config-if)# ip igmp startup-query-count 4
```

The following example shows how to remove the IGMP startup query count for VLAN interface vlan3:

```
          awplus# configure terminal
  awplus(config)# interface vlan3
awplus(config-if)# no ip igmp startup-query-count
```

Related Commands ip igmp last-member-query-count
ip igmp startup-query-interval

ip igmp startup-query-interval

Use this command to configure the IGMP startup query interval for a VLAN interface in Interface Configuration mode. The IGMP startup query interval is the amount of time in seconds between successive IGMP General Query messages sent by a querier during startup. The default IGMP startup query interval is one quarter of the IGMP query interval value.

Use the **no** variant of this command to remove the configured IGMP startup query interval for a VLAN interface in Interface Configuration mode.

Syntax

```
ip igmp startup-query-interval <startup-query-interval>

no ip igmp startup-query-interval
```

Parameter	Description
<startup-query-interval>	Specify the IGMP startup query interval for a VLAN interface in Interface Configuration mode in the range of <2-1800> seconds to be one quarter of the IGMP query interval value.

Default The default IGMP startup query interval is one quarter of the IGMP query interval value.

Note The IGMP startup query interval must be one quarter of the IGMP query interval.

Mode Interface Configuration for a VLAN interface.

Examples The following example shows how to configure the IGMP startup query interval to 15 seconds for VLAN interface `vlan2` to be one quarter of the IGMP query interval value of 60 seconds:

```
          awplus#  configure terminal
  awplus(config)#  interface vlan2
awplus(config-if)#  ip igmp startup-query-interval 15
awplus(config-if)#  ip igmp query-interval 60
```

The following example shows how to remove the IGMP startup query interval for VLAN interface `vlan2`:

```
          awplus#  configure terminal
  awplus(config)#  interface vlan2
awplus(config-if)#  no ip igmp startup-query-interval
```

Related Commands ip igmp last-member-query-interval
ip igmp query-interval
ip igmp startup-query-count

ip igmp version

Use this command to set the current IGMP version (IGMP version 1, 2 or 3) on an interface.

Use the **no** variant of this command to return to the default version.

Syntax

```
ip igmp version <1-3>
no ip igmp version
```

Parameter	Description
<1-3>	IGMP protocol version number

Default The default IGMP protocol version number is 3.

Mode Interface Configuration for a VLAN interface.

Usage This command applies to VLAN interfaces configured for IGMP.

Example

```
          awplus# configure terminal
  awplus(config)# interface vlan5
awplus(config-if)# ip igmp version 2
```

Validation Commands show ip igmp interface

show debugging igmp

Use this command to display the IGMP debugging options set.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax `show debugging igmp`

Mode User Exec and Privileged Exec

Example To display the IGMP debugging options set, enter the command:

```
awplus# show debugging igmp
```

Output Figure 29-1: Example output from the **show debugging igmp** command

```
IGMP Debugging status:
  IGMP Decoder debugging is on
  IGMP Encoder debugging is on
  IGMP Events debugging is on
  IGMP FSM debugging is on
  IGMP Tree-Info-Base (TIB) debugging is on
```

Related Commands debug igmp

show ip igmp groups

Use this command to display the multicast groups with receivers directly connected to the router, and learned through IGMP.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax

```
show ip igmp groups [<ip-address>|<interface> detail]
```

Parameter	Description
<ip-address>	Address of the multicast group, entered in the form A.B.C.D.
<interface>	Interface name for which to display local information.

Mode User Exec and Privileged Exec

Example The following command displays local-membership information for all ports in all interfaces:

```
awplus# show ip igmp groups
```

Output Figure 29-2: Example output from the **show ip igmp groups** command

```
IGMP Connected Group Membership
Group Address    Interface       Uptime    Expires   Last
Reporter
224.0.1.1        port1.0.1       00:00:09  00:04:17  10.10.0.82
224.0.1.24       port1.0.2       00:00:06  00:04:14  10.10.0.84
224.0.1.40       port1.0.3       00:00:09  00:04:15  10.10.0.91
224.0.1.60       port1.0.3       00:00:05  00:04:15  10.10.0.7
224.100.100.100  port1.0.1       00:00:11  00:04:13  10.10.0.91
228.5.16.8       port1.0.3       00:00:11  00:04:16  10.10.0.91
228.81.16.8      port1.0.7       00:00:05  00:04:15  10.10.0.91
228.249.13.8     port1.0.3       00:00:08  00:04:17  10.10.0.91
235.80.68.83     port1.0.11      00:00:12  00:04:15  10.10.0.40
239.255.255.250  port1.0.3      00:00:12  00:04:15  10.10.0.228
239.255.255.254  port1.0.12      00:00:08  00:04:13  10.10.0.84
```

Table 29-1: Parameters in the output of the **show ip igmp groups** command

Parameter	Description
`Group Address`	Address of the multicast group.
`Interface`	Port through which the group is reachable.
`Uptime`	The time in weeks, days, hours, minutes, and seconds that this multicast group has been known to the device.
`Expires`	Time (in hours, minutes, and seconds) until the entry expires.
`Last Reporter`	Last host to report being a member of the multicast group.

show ip igmp interface

Use this command to display the state of IGMP, IGMP Proxy service, and IGMP Snooping for a specified VLAN, or all VLANs. IGMP is shown as Active or Disabled in the show output.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax `show ip igmp interface [<interface>]`

Parameter	Description
<interface>	The name of the VLAN interface.

Mode User Exec and Privileged Exec

Examples The following output shows IGMP interface status for **vlan2** (with IGMP Snooping enabled):

```
awplus#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
awplus(config)#interface vlan2
awplus(config-if)#ip igmp snooping
awplus(config-if)#exit
awplus(config)#exit
awplus#show ip igmp interface vlan2
Interface vlan2 (Index 202)
 IGMP Disabled, Inactive, Version 3 (default)
 IGMP interface has 0 group-record states
 IGMP activity: 0 joins, 0 leaves
 IGMP robustness variable is 2
 IGMP last member query count is 2
 IGMP query interval is 125 seconds
 IGMP query holdtime is 500 milliseconds
 IGMP querier timeout is 255 seconds
 IGMP max query response time is 10 seconds
 Last member query response interval is 1000 milliseconds
 Group Membership interval is 260 seconds
 Strict IGMPv3 ToS checking is disabled on this interface
 Source Address checking is enabled
 IGMP Snooping is globally enabled
 IGMP Snooping query solicitation is globally disabled
  Num. query-solicit packets: 57 sent, 0 recvd
 IGMP Snooping is enabled on this interface
 IGMP Snooping fast-leave is not enabled
 IGMP Snooping querier is not enabled
 IGMP Snooping report suppression is enabled
awplus#
```

The following output shows IGMP interface status for **vlan2** (with IGMP Snooping disabled):

```
awplus#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
awplus(config)#interface vlan2
awplus(config-if)#no ip igmp snooping
awplus(config-if)#exit
awplus(config)#exit
awplus#show ip igmp interface vlan2
Interface vlan2 (Index 202)
 IGMP Disabled, Inactive, Version 3 (default)
 IGMP interface has 0 group-record states
 IGMP activity: 0 joins, 0 leaves
 IGMP robustness variable is 2
 IGMP last member query count is 2
 IGMP query interval is 125 seconds
 IGMP query holdtime is 500 milliseconds
 IGMP querier timeout is 255 seconds
 IGMP max query response time is 10 seconds
 Last member query response interval is 1000 milliseconds
 Group Membership interval is 260 seconds
 Strict IGMPv3 ToS checking is disabled on this interface
 Source Address checking is enabled
 IGMP Snooping is globally enabled
 IGMP Snooping query solicitation is globally disabled
  Num. query-solicit packets: 57 sent, 0 recvd
 IGMP Snooping is not enabled on this interface
 IGMP Snooping fast-leave is not enabled
 IGMP Snooping querier is not enabled
 IGMP Snooping report suppression is enabled
awplus#
```

The following command displays the IGMP interface status and Query Solicitation for **vlan3**:

```
awplus#show ip igmp interface vlan3
Interface vlan3 (Index 203)
 IGMP Enabled, Active, Querier, Version 3 (default)
 Internet address is 192.168.9.1
 IGMP interface has 256 group-record states
 IGMP activity: 51840 joins, 0 leaves
 IGMP robustness variable is 2
 IGMP last member query count is 2
 IGMP query interval is 125 seconds
 IGMP query holdtime is 500 milliseconds
 IGMP querier timeout is 250 seconds
 IGMP max query response time is 1 seconds
 Last member query response interval is 1000 milliseconds
 Group Membership interval is 251 seconds
 Strict IGMPv3 ToS checking is disabled on this interface
 IGMP Snooping is globally enabled
 IGMP Snooping query solicitation is globally enabled
  Num. query-solicit packets: 1 sent, 10 recvd
 IGMP Snooping is enabled on this interface
 IGMP Snooping fast-leave is not enabled
 IGMP Snooping querier is not enabled
 IGMP Snooping report suppression is enabled
awplus#
```

Note Query Solicitation status information is highlighted in **bold** in the above output.

Use the **show ip igmp interface** command to validate that Query Solicitation is enabled and to show the number of query-solicit message packets sent and received on a VLAN.

Related Commands

clear ip igmp
clear ip igmp group
clear ip igmp interface
ip igmp
ip igmp last-member-query-count
ip igmp last-member-query-interval
ip igmp querier-timeout
ip igmp query-holdtime
ip igmp query-interval
ip igmp query-max-response-time
ip igmp robustness-variable
ip igmp snooping
ip igmp snooping fast-leave
ip igmp snooping querier
ip igmp snooping report-suppression
ip igmp snooping tcn query solicit
ip igmp version

show ip igmp proxy

Use this command to display the state of IGMP Proxy services for a specified interface or for all interfaces.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax

```
show ip igmp proxy

show ip igmp proxy groups [detail]

show ip igmp proxy groups <multicast-group> [detail]

show ip igmp proxy groups <vlan> [detail]

show ip igmp proxy groups <vlan> <multicast-group> [detail]
```

Parameter	Description
`groups`	Specify IGMP proxy group membership information.
`detail`	Specify detailed IGMPv3 source information.
<vlan>	Specify the name of a single VLAN interface, for example **vlan1**.
<multicast-group>	Specify the IPv4 address in of the multicast group, in the format A.B.C.D.

Mode User Exec and Privileged Exec

Example To display the state of IGMP Proxy services for all interfaces, enter the command:

```
awplus# show ip igmp proxy
```

To display the state of IGMP Proxy services for VLAN interface **vlan1**, enter the command:

```
awplus# show ip igmp proxy groups vlan1
```

To display the detailed state of IGMP Proxy services for VLAN interface **vlan1**, enter the command:

```
awplus# show ip igmp proxy groups vlan1 detail
```

Related Commands ip igmp proxy-service

show ip igmp snooping mrouter

Use this command to display the multicast router ports, both static and dynamic, in a VLAN.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax

```
show ip igmp snooping mrouter [interface <interface>]
```

Parameter	Description
`interface`	A specific interface.
`<interface>`	The name of the VLAN interface.

Mode User Exec and Privileged Exec

Example To show all multicast router interfaces, use the command:

```
awplus# show ip igmp snooping mrouter
```

To show the multicast router interfaces in `vlan1`, use the command:

```
awplus# show ip igmp snooping mrouter interface vlan1
```

Output Figure 29-3: Example output from the **show ip igmp snooping mrouter** command

```
VLAN     Interface    Static/Dynamic
1        port1.0.5    Statically configured
200      port1.0.2    Statically configured
```

Figure 29-4: Example output from the **show ip igmp snooping mrouter interface vlan1** command

```
VLAN     Interface    Static/Dynamic
1        port1.0.5    Statically configured
```

Related Commands ip igmp snooping mrouter

show ip igmp snooping routermode

Use this command to display the current routermode and the list of IP addresses set as router multicast addresses from the ip igmp snooping routermode command.

For information on output options, see “Controlling “show” Command Output” on page 1.41.

Syntax `show ip igmp snooping routermode`

Mode User Exec and Privileged Exec

Example To show the routermode and the list of router multicast addresses, use the command:

```
awplus# show ip igmp snooping routermode
```

Output Figure 29-5: Example output from the **show ip igmp snooping routermode** command

```
Router mode.............Def

Reserved multicast address
         224.0.0.1
         224.0.0.2
         224.0.0.4
         224.0.0.5
         224.0.0.6
         224.0.0.9
         224.0.0.13
         224.0.0.15
          224.0.0.24
```

Related Commands ip igmp snooping routermode

show ip igmp snooping statistics

Use this command to display IGMP Snooping statistics data.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax

```
show ip igmp snooping statistics interface <interface-range>
    [group [<ip-address>]]
```

Parameter	Description
<ip-address>	Optionally specify the address of the multicast group, entered in the form A.B.C.D.
<interface>	Specify the name of the VLAN interface or interface range.

Mode User Exec and Privileged Exec

Example To display IGMP statistical information for **vlan1** and **vlan2**, use the command:

```
awplus# show ip igmp snooping statistics interface
        vlan1-vlan2
```

Output Figure 29-6: Example output from the **show ip igmp snooping statistics** command

```
IGMP Snooping statistics for vlan1
Interface:      port1.0.3
Group:          224.1.1.1
Uptime:         00:00:09
Group mode:     Exclude (Expires: 00:04:10)
Last reporter:  10.4.4.5
Source list is empty
IGMP Snooping statistics for vlan2
Interface:      port1.0.4
Group:          224.1.1.2
Uptime:         00:00:19
Group mode:     Exclude (Expires: 00:05:10)
Last reporter:  10.4.4.6
Source list is empty
```

undebug igmp

This command applies the functionality of the no debug igmp command on page 29.5.

Chapter 30: MLD Snooping Introduction and Commands

Introduction

Multicast Listener Discovery (MLD) is used to exchange membership status information between IPv6 routers that support multicasting and members of multicast groups on a network segment. Host membership in a multicast group is reported by individual member hosts, and membership status is periodically polled by multicast routers. For a general overview of multicasting, see Chapter 49, Multicast Introduction and Commands.

MLD is defined in RFC 2710, "Multicast Listener Discovery (MLD) for IPv6."

Note IPv6 is only supported in the stand-alone mode. It is not supported in VCStack configurations.

MLD Snooping

MLD Snooping is a feature whereby a Layer 2 switch listens to or "snoops" the MLD messages passing through the switch or from member hosts and multicast routers. The purpose of MLD Snooping is to provide efficient Layer 2 multicast forwarding, by sending only to hosts that have expressed an interest in receiving the multicast data.

Hosts express an interest in receiving multicast data for a given multicast group by sending an MLD join message. Without MLD Snooping, if one host expresses an interest in getting multicast data for a given group, by sending an MLD join for the multicast group, then all hosts connected to the same VLAN will also receive the multicast data. This wastes bandwidth on the switch ports connected to the host that are not interested in receiving the multicast data. Snooping takes note of exactly which ports have received joins for a given group, and send that group only to those ports.

MLD Snooping is enabled by default globally for the switch. It can be enabled and disabled on a per-VLAN basis.

For MLD Snooping to operate, both IGMP Snooping and MLD Snooping must be enabled globally on the switch. By default, IGMP Snooping is also enabled globally. To enable IGMP Snooping if it has been disabled, use the ip igmp snooping command on page 29.22 in Global Configuration mode.

MLD Snooping makes a distinction between Member ports, which are ports connected to members hosts, and Router ports, which are ports connected to, or directed towards, a Layer 3 router or a Layer 3 switch.

Command List

This chapter provides an alphabetical reference of configuration, clear, and show commands related to MLD Snooping.

clear ipv6 mld

Use this command to clear all MLD local memberships on all interfaces.

Syntax

```
clear ipv6 mld
```

Mode Privileged Exec

Usage This command applies to groups learned by MLD Snooping.

Example

```
awplus# clear ipv6 mld
```

Related Commands clear ipv6 mld group
clear ipv6 mld interface

clear ipv6 mld group

Use this command to clear MLD specific local-membership(s) on all interfaces, for a particular group.

Note This command is only supported in the stand-alone mode. It is not supported in VCStack configurations.

Syntax

```
clear ipv6 mld group {*|<ipv6-address>}
```

Parameter	Description
*	Clears all groups on all interfaces. This is an alias to the clear ipv6 mld command.
<ipv6-address>	Specify the group address for which MLD local-memberships are to be cleared from all interfaces. Specify the IPv6 multicast group address in the format in the format X:X::X:X.

Mode Privileged Exec

Usage This command applies to groups learned by MLD Snooping.

Example

```
awplus# clear ipv6 mld group *
```

Related Commands

clear ipv6 mld
clear ipv6 mld interface

clear ipv6 mld interface

Use this command to clear MLD interface entries.

Note This command is only supported in the stand-alone mode. It is not supported in VCStack configurations.

Syntax

```
clear ipv6 mld interface <interface>
```

Parameter	Description
<interface>	Specifies name of the interface; all groups learned from this interface are deleted.

Mode Privileged Exec

Usage This command applies to interfaces configured for MLD Snooping.

Example

```
awplus# clear ipv6 mld interface vlan2
```

Related Commands

clear ipv6 mld
clear ipv6 mld group

debug mld

Use this command to enable all MLD debugging modes, or a specific MLD debugging mode.

Use the **no** variant of this command to disable all MLD debugging modes, or a specific MLD debugging mode.

Note This command is only supported in the stand-alone mode. It is not supported in VCStack configurations.

Syntax

```
debug mld {all|decode|encode|events|fsm|tib}

no debug mld {all|decode|encode|events|fsm|tib}
```

Parameter	Description
`all`	Debug all MLD.
`decode`	Debug MLD decoding.
`encode`	Debug MLD encoding.
`events`	Debug MLD events.
`fsm`	Debug MLD Finite State Machine (FSM).
`tib`	Debug MLD Tree Information Base (TIB).

Mode Privileged Exec and Global Configuration

Examples

```
        awplus# configure terminal
awplus(config)# debug mld all

        awplus# configure terminal
awplus(config)# debug mld decode

        awplus# configure terminal
awplus(config)# debug mld encode

        awplus# configure terminal
awplus(config)# debug mld events
```

ipv6 mld access-group

Use this command to control the multicast local-membership groups learned on an interface.

Use the **no** variant of this command to disable this access control.

Note This command is only supported in the stand-alone mode. It is not supported in VCStack configurations.

Syntax

```
ipv6 mld access-group <access-list-name>

no ipv6 mld access-group
```

Parameter	Description
<access-list-name>	Standard IPv6 access-list name.

Default No access list is configured by default.

Mode Interface Configuration for a specified VLAN interface or a range of VLAN interfaces.

Usage This command applies to groups learned by MLD Snooping.

Example In the following example, `vlan2` will only accept MLD joins for groups in the range `ff1e:0db8:0001::/64`:

```
         awplus#  configure terminal
 awplus(config)#  ipv6 access-list standard group1 permit
                  ff1e:0db8:0001::/64
 awplus(config)#  interface vlan2
awplus(config-if)#  ipv6 mld access-group group1
```

In the following example, `vlan2-vlan4` will only accept MLD joins for groups in the range `ff1e:0db8:0001::/64`:

```
         awplus#  configure terminal
 awplus(config)#  ipv6 access-list standard group1 permit
                  ff1e:0db8:0001::/64
 awplus(config)#  interface vlan2-vlan4
awplus(config-if)#  ipv6 mld access-group group1
```

ipv6 mld limit

Use this command to configure a limit on the maximum number of group memberships that may be learned. The limit may be set for the switch as a whole, or for a specific interface.

Once the specified group membership limit is reached, all further local-memberships will be ignored.

Optionally, an exception access-list can be configured to specify the group-address(es) that are exempted from being subject to the limit.

Use the **no** variant of this command to unset the limit and any specified exception access-list.

Note This command is only supported in the stand-alone mode. It is not supported in VCStack configurations.

Syntax

```
ipv6 mld limit <limitvalue> [except <access-list-name>]

no ipv6 mld limit
```

Parameter	Description
<limitvalue>	<2-512> Maximum number of group membership states.
<access-list-name>	Standard IPv6 access-list name that defines multicast groups which are exempted from being subject to the configured limit.

Default The default limit, which is reset by the **no** variant of this command, is the same as maximum number of group membership entries that can be learned with the **ipv6 mld limit** command.

The default limit of group membership entries that can be learned is 512 entries.

Mode Global Configuration and Interface Configuration for a specified VLAN interface or a range of VLAN interfaces.

Usage This command applies to interfaces learned by MLD Snooping.

If this command is issued on multiple VLAN interfaces, the limits apply individually to each of those interfaces.

Examples The following example configures an MLD limit of 100 group-memberships across all VLAN interfaces on which MLD is enabled, and excludes groups in the range `ff1e:0db8:0001::/64` from this limitation:

```
        awplus# configure terminal

awplus(config)# ipv6 access-list standard v6grp permit
                ff1e:0db8:0001::/64

awplus(config)# ipv6 mld limit 100 except v6grp
```

The following example configures an MLD limit of 100 group-membership states on vlan2:

```
           awplus# configure terminal
   awplus(config)# interface vlan2
awplus(config-if)# ipv6 mld limit 100
```

The following example configures an MLD limit of 100 group-membership states on vlan2-vlan4:

```
           awplus# configure terminal
   awplus(config)# interface vlan2-vlan4
awplus(config-if)# ipv6 mld limit 100
```

Related Commands show ipv6 mld groups

ipv6 mld snooping

Use this command to enable MLD Snooping. When this command is issued in the Global Configuration mode, MLD Snooping is enabled globally for the switch. When this command is issued in Interface mode for a VLAN then MLD Snooping is enabled for the specified VLAN. Note that MLD Snooping is enabled on the VLAN only if it is enabled globally and on the VLAN.

Use the **no** variant of this command to globally disable MLD Snooping in Global Configuration mode, or for the specified VLAN interface in Interface mode.

Note This command is only supported in the stand-alone mode. It is not supported in VCStack configurations.

Syntax

```
ipv6 mld snooping

no ipv6 mld snooping
```

Default By default, MLD Snooping is enabled both globally and on all VLANs.

Mode Global Configuration and Interface Configuration for a specified VLAN interface or a range of VLANs.

Usage For MLD Snooping to operate on particular VLAN interfaces, it must be enabled both globally by using this command in Global Configuration mode, and on individual VLAN interfaces by using this command in Interface Configuration mode (Both are enabled by default).

Both IGMP Snooping and MLD Snooping must be enabled globally on the switch for MLD Snooping to operate. IGMP Snooping is also enabled by default. To enable it if it has been disabled, use the ip igmp snooping command on page 29.22 in Global Configuration mode.

Examples To configure MLD Snooping on `vlan2`, enter the following commands:

```
          awplus#  configure terminal
  awplus(config)#  interface vlan2
awplus(config-if)#  ipv6 mld snooping
```

To configure MLD Snooping on `vlan2-vlan4`, enter the following commands:

```
          awplus#  configure terminal
  awplus(config)#  interface vlan2-vlan4
awplus(config-if)#  ipv6 mld snooping
```

To disable MLD Snooping for `vlan2`, enter the following commands:

```
         awplus#  configure terminal
 awplus(config)#  interface vlan2
 awplus(config)#  no ipv6 mld snooping
```

To disable MLD Snooping for vlan2-vlan4, enter the following commands:

```
        awplus# configure terminal
awplus(config)# interface vlan2-vlan4
awplus(config)# no ipv6 mld snooping
```

To configure MLD Snooping globally for the switch, enter the following commands:

```
        awplus# configure terminal
awplus(config)# ipv6 mld snooping
```

To disable MLD Snooping globally for the switch, enter the following commands:

```
        awplus# configure terminal
awplus(config)# no ipv6 mld snooping
```

Related Commands ip igmp snooping

ipv6 mld snooping fast-leave

Use this command to enable MLD Snooping fast-leave processing. Fast-leave processing is analogous to immediate-leave processing; the MLD group-membership is removed as soon as an MLD leave group message is received, without sending out a group-specific query.

Use the **no** variant of this command to disable fast-leave processing.

Note This command is only supported in the stand-alone mode. It is not supported in VCStack configurations.

There is a 100 MLD interface limit when applying this command to multiple VLANs.

Syntax

```
ipv6 mld snooping fast-leave
```

```
no ipv6 mld snooping fast-leave
```

Default MLD Snooping fast-leave processing is disabled.

Mode Interface Configuration for a specified VLAN interface or a range of VLAN interfaces.

Usage This MLD Snooping command can only be configured on VLAN interfaces.

Example This example shows how to enable fast-leave processing on `vlan2`.

```
          awplus# configure terminal
  awplus(config)# interface vlan2
awplus(config-if)# ipv6 mld snooping fast-leave
```

This example shows how to enable fast-leave processing on `vlan2-vlan4`.

```
          awplus# configure terminal
  awplus(config)# interface vlan2-vlan4
awplus(config-if)# ipv6 mld snooping fast-leave
```

ipv6 mld snooping mrouter

Use this command to statically configure the specified port as a Multicast Router interface for MLD Snooping within the specified VLAN.

Use the **no** variant of this command to remove the static configuration of the interface as a Multicast Router interface.

Note This command is only supported in the stand-alone mode. It is not supported in VCStack configurations.

There is a 100 MLD interface limit when applying this command to multiple VLANs.

Syntax

```
ipv6 mld snooping mrouter interface <port>

no ipv6 mld snooping mrouter interface <port>
```

Parameter	Description
<port>	Specify the name of the port.

Mode Interface Configuration for a specified VLAN interface or a range of VLAN interfaces.

Usage This MLD Snooping command statically configures a switch port as a Multicast Router interface.

Example This example shows how to specify the next-hop interface to the multicast router for `vlan2`.

```
          awplus#  configure terminal
   awplus(config)#  interface vlan2
awplus(config-if)#  ipv6 mld snooping mrouter interface
                    port1.0.5
```

This example shows how to specify the next-hop interface to the multicast router for `vlan2-vlan4`.

```
          awplus#  configure terminal
   awplus(config)#  interface vlan2-vlan4
awplus(config-if)#  ipv6 mld snooping mrouter interface
                    port1.0.5
```

ipv6 mld snooping report-suppression

Use this command to enable report suppression from hosts for Multicast Listener Discovery version 1 (MLDv1) on a VLAN in interface configuration mode.

Use the **no** variant of this command to disable report suppression on a VLAN in interface configuration mode.

Note This command is only supported in the stand-alone mode. It is not supported in VCStack configurations.

There is a 100 MLD interface limit when applying this command to multiple VLANs.

Syntax

```
ipv6 mld snooping report-suppression

no ipv6 mld snooping report-suppression
```

Default Report suppression does not apply to MLDv2, and is turned on by default for MLDv1 reports.

Mode Interface Configuration for a specified VLAN interface or a range of VLAN interfaces.

Usage This MLD Snooping command can only be configured on VLAN interfaces.

MLDv1 Snooping maybe configured to suppress reports from hosts. When a querier sends a query, only the first report for particular set of group(s) from a host will be forwarded to the querier by the MLD Snooping switch. Similar reports (to the same set of groups) from other hosts, which would not change group memberships in the querier, will be suppressed by the MLD Snooping switch to prevent 'flooding' of query responses.

Examples This example shows how to enable report suppression for MLD reports on `vlan2`.

```
          awplus#  configure terminal
  awplus(config)#  interface vlan2
awplus(config-if)#  ipv6 mld snooping report-suppression
```

This example shows how to enable report suppression for MLD reports on `vlan2-vlan4`.

```
          awplus#  configure terminal
  awplus(config)#  interface vlan2-vlan4
awplus(config-if)#  ipv6 mld snooping report-suppression
```

This example shows how to disable report suppression for MLD reports on `vlan2`:

```
          awplus#  configure terminal
  awplus(config)#  interface vlan2
awplus(config-if)#  no ipv6 mld snooping report-suppression
```

This example shows how to disable report suppression for MLD reports on `vlan2-vlan4`:

```
          awplus# configure terminal
  awplus(config)# interface vlan2-vlan4
awplus(config-if)# no ipv6 mld snooping report-suppression
```

ipv6 mld static-group

Use this command to statically configure IPv6 group membership entries on an interface. To statically add only a group membership, do not specify any parameters.

Use the **no** variant of this command to delete static group membership entries.

Note This command is only supported in the stand-alone mode. It is not supported in VCStack configurations.

Syntax

```
ipv6 mld static-group <ipv6-group-address>
   [source <ipv6-source-address>] [interface <port>]

no ipv6 mld static-group <ipv6-group-address>
   [source <ipv6-source-address>] [interface <port>]
```

Parameter	Description
<ipv6-group-address>	Specify a standard IPv6 Multicast group address to be configured as a static group member. The IPv6 address uses the format X:X::X:X.
<ipv6-source-address>	Optional. Specify a standard IPv6 source address to be configured as a static source from where multicast packets originate. The IPv6 address uses the format X:X::X:X.
<port>	Optional. Physical interface. This parameter specifies a physical port. If this parameter is used, the static configuration is applied to just to that physical interface. If this parameter is not used, the static configuration is applied on all ports in the VLAN.

Mode Interface Configuration for a specified VLAN interface.

Usage This command applies to MLD Snooping on a VLAN interface to statically add groups and/or source records.

Example The following examples show how to statically add group and/or source records for MLD Snooping on `vlan2`:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ipv6 mld static-group ff1e::10
```

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ipv6 mld static-group ff1e::10 source
                   fe80::2fd:6cff:fe1c:b
```

```
awplus#  configure terminal
awplus(config)#  interface vlan2
awplus(config-if)#  ipv6 mld static-group ff1e::10
```

```
awplus#  configure terminal
awplus(config)#  interface vlan2
awplus(config-if)#  ipv6 mld static-group ff1e::10 interface
                    port1.0.8
```

```
awplus#  configure terminal
awplus(config)#  interface vlan2
awplus(config-if)#  ipv6 mld static-group ff1e::10 source
                    fe80::2fd:6cff:fe1c:b interface port1.0.8
```

```
awplus#  configure terminal
awplus(config)#  interface vlan2
awplus(config-if)#  ipv6 mld static-group ff1e::10 interface
                    port1.0.8
```

show ipv6 mld groups

Use this command to display the multicast groups with receivers directly connected to the router, and learned through MLD.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Note This command is only supported in the stand-alone mode. It is not supported in VCStack configurations.

Syntax

```
show ipv6 mld groups [<ipv6-address> |<interface> detail]
```

Parameter	Description
<ipv6-address>	Optional. Specify Address of the multicast group in format X:X::X:X.
<interface>	Optional. Specify the Interface name for which to display local information.

Mode User Exec and Privileged Exec

Example The following command displays local-membership information for all interfaces:

```
awplus# show ipv6 mld groups
```

Output Figure 30-1: Example output from the **show ipv6 mld groups** command

```
MLD Connected Group Membership
Group Address     Interface     Uptime      Expires    Last
Reporter
ff1e::10          ge10          00:03:16    00:01:09
fe80::202:b3ff:fef0:79d8
```

show ipv6 mld interface

Use this command to display the state of MLD Snooping for a specified interface, or all interfaces.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Note This command is only supported in the stand-alone mode. It is not supported in VCStack configurations.

There is a 100 MLD interface limit when applying this command to multiple VLANs.

Syntax

```
show ipv6 mld interface [<interface>]
```

Parameter	Description
<interface>	Interface name.

Mode

User Exec and Privileged Exec

Example

The following command displays MLD interface status on all interfaces enabled for MLD:

```
awplus# show ipv6 mld interface
```

Output

Figure 30-2: Example output from the **show ipv6 mld interface** command

```
Interface vlan1 (Index 2)
 MLD Enabled, Active, Querier, Version 2 (default)
 Internet address is fe80::2fd:6cff:fe1c:b
 MLD interface has 0 group-record states
 MLD activity: 0 joins, 0 leaves
 MLD query interval is 125 seconds
 MLD querier timeout is 255 seconds
 MLD max query response time is 10 seconds
 Last member query response interval is 1000 milliseconds
 Group Membership interval is 260 seconds
```

show ipv6 mld snooping mrouter

Use this command to display the multicast router interfaces, both configured and learned, in a VLAN.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Note This command is only supported in the stand-alone mode. It is not supported in VCStack configurations.

There is a 100 MLD interface limit when applying this command to multiple VLANs.

Syntax `show ipv6 mld snooping mrouter <interface>`

Parameter	Description
<interface>	The name of the VLAN interface.

Mode User Exec and Privileged Exec

Example The following command displays the multicast router interfaces in `vlan2`:

```
awplus# show ipv6 mld snooping mrouter vlan2
```

Output Figure 30-3: Example output from the **show ipv6 mld snooping mrouter** command

```
VLAN     Interface     Static/Dynamic
2        port1.0.2
2        port1.0.3
```

show ipv6 mld snooping statistics

Use this command to display MLD Snooping statistics data.

For information on output options, see “Controlling “show” Command Output” on page 1.41.

Note This command is only supported in the stand-alone mode. It is not supported in VCStack configurations.

There is a 100 MLD interface limit when applying this command to multiple VLANs.

Syntax

```
show ipv6 mld snooping statistics interface <interface>
```

Parameter	Description
<*interface*>	The name of the VLAN interface.

Mode User Exec and Privileged Exec

Example The following command displays MLDv2 statistical information for `vlan2`:

```
awplus# show ipv6 mld snooping statistics interface vlan2
```

Output Figure 30-4: Example output from the **show ipv6 mld snooping statistics** command

```
Interface:      vlan2
Group:          ff1e::10
Uptime:         00:00:13
Group mode:     Include
Last reporter:  fe80::202:b3ff:fef0:79d8
Group source list:
  Source Address         Uptime    v2 Exp    Fwd  Flags
  7ffe::4                00:00:13  00:04:06  Yes  R
```

Part 5: Access and Security

Chapter 31: Access Control Lists Introduction

Introduction

This chapter describes Access Control Lists (ACLs), and general ACL configuration information.

See Chapter 32, IPv4 Hardware Access Control List (ACL) Commands for detailed command information and command examples about IPv4 hardware ACLs that are applied directly to interfaces.

See Chapter 33, IPv4 Software Access Control List (ACL) Commands and Chapter 34, IPv6 Software Access Control List (ACL) Commands for detailed command information and command examples about IPv4 and IPv6 software ACLs as applied to Routing and Multicasting.

See all relevant Routing commands and configurations in "Layer Three, Switching and Routing" and all relevant Multicast commands and configurations in "Multicast Applications".

Overview

An Access Control List (ACL) is one filter, or a sequence of filters, that are applied to an interface to either block, pass, or when using QoS, apply priority to, packets that match the filter definitions. ACLs are used to restrict network access by hosts and devices and to limit network traffic.

An ACL contains an ordered list of filters. Each filter specifies either permit or deny and a set of conditions the packet must satisfy in order to match the filter. The meaning of permit or deny entries depends on the context in which the ACL is used - either on an inbound or an outbound interface.

When a packet is received on an interface, the switch compares fields in the packet against filters in the ACL to check whether the packet has permission to be forwarded, based on the filter properties. The first match determines whether the switch accepts or rejects the packets. If no entries match, the switch rejects the packets. If there are no restrictions, the switch forwards the packets.

Because filters in an ACL are applied sequentially and their action stops at the first match, it is very important that you apply the filters in the correct order. For example you might want to pass all traffic from VLAN 4 except for that arriving from two selected addresses A and B. Setting up a filter that first passes all traffic from VLAN 4 then denies traffic from addresses A and B will not filter out traffic from A and B if they are members VLAN 4. To ensure that the traffic from A and B is always blocked you should first apply the filter to block traffic from A and B, then apply the filter to allow all traffic from VLAN 4.

You can assign sequence numbers to filters. See "ACL Filter Sequence Numbers" on page 31.14 for more information.

ACL Rules

- The source or destination address or the protocol of each packet being filtered are tested against the filters in the ACL, one condition at a time (for a permit or a deny filter).
- If a packet does not match a filter then the packet is checked against the next filter in the ACL.
- If a packet and a filter match, the subsequent filters in the ACL are not checked and the packet is permitted or denied as specified in the matched filter.
- The first filter that the packet matches determines whether the packet is permitted or denied. After the first match, no subsequent filters are considered.
- If the ACL denies the address or protocol then the software discards the packet.
- For software ACLs, if no filters match then the packet is dropped.
- For hardware ACLs, if no filters match then the packet is forwarded.
- Checking stops after the first match, so the order of the filters in the ACL is critical. The same permit or deny filter specified in a different order could result in a packet being passed in one situation and denied in another situation.
- One ACL per interface, per protocol, per direction is allowed. However, each ACL assigned per interface, per protocol, per direction may also have multiple filters.
- For inbound ACLs, a permit filter continues to process the packet after receiving it on an inbound interface, and a deny filter discards the packet.

ACL Source and Destination Addresses

Configure source addresses in ACL filters to filter packets coming **from** specified networking devices or hosts. Configure destination addresses in ACL filters to filter packets going **to** specified networking devices or hosts.

ACL Reverse Masking

ACLs uses reverse masking, also referred to as wildcard masking, to indicate to the switch whether to check or ignore corresponding IP address bits when comparing the address bits in an ACL filter to a packet being submitted to the ACL.

Reverse masking for IP address bits specify how the switch treats the corresponding IP address bits. A reverse mask is also called an inverted mask because a 1 and 0 mean the opposite of what they mean in a subnet or a network mask.

- A reverse mask bit 0 means check the corresponding bit value.
- A reverse mask bit 1 means ignore the corresponding bit value.

Hardware and Software ACL Types

Access Control Lists (ACLs) used in AlliedWare Plus^{TM} are separated into two different types, **Software ACLs** and **Hardware ACLs**. You can define both types as either named or numbered.

Note The filtering principles applied to software ACLs (those in the range 1 to 2699) are different to those applied to hardware ACLs (those in the range 3000 to 4699).

Software ACLs will **deny** access unless **explicitly permitted** by an ACL action. Hardware ACLs will **permit** access unless **explicitly denied** by an ACL action.

Numbered ACLs (for Hardware and Software ACLs)

Numbered ACLs are assigned an ACL number within the range 1 to 4699. ACL numbers are grouped into ranges, where each range denotes a specific functionality. The following table shows the number ranges and functionality that your switch supports.

Table 31-1: ACL Numeric Ranges and Functionality

ACL Number Range	Function
1 to 99	IP standard ACL [1]
100 to 199	IP extended ACL [1]
1300 to 1999	IP standard expanded ACL [1]
2000 to 2699	IP extended expanded ACL [1]
3000 to 3699	Hardware IP ACL
4000 to 4699	Hardware MAC ACL

1. Software ACLs that use either the ranges 1-99, 100-199, 1300-1999, 2000-2699, or are named ACLs (that use the standard or extended keyword followed by a text string), are used in features such as SNMP.

Hardware ACLs

These ACL types are applied directly to an interface, or are used for QoS classifications. They use the following ranges:

- 3000-3699 for Hardware IP ACLs
- 4000-4699 for Hardware MAC ACLs
- named hardware IPv4 ACLs

See Chapter 32, IPv4 Hardware Access Control List (ACL) Commands for detailed command information and command examples about IPv4 hardware ACLs that are applied directly to interfaces.

Software ACLs

These ACLs types can be either named ACLs, using the standard or extended keyword followed by a text string, or they can use the following ranges:

- 1-99 (IP standard ACL range)
- 100-199 (IP extended ACL range)
- 1300-1999 (IP standard expanded ACL range)
- 2000-2699 (IP extended expanded ACL range)
- named standard IPv4 ACLs
- named extended IPv4 ACLs
- named standard IPv6 ACLs
- named extended IPv6 ACLs

Software ACLs are used in features such as SNMP and PIM.

See Chapter 33, IPv4 Software Access Control List (ACL) Commands and Chapter 34, IPv6 Software Access Control List (ACL) Commands for detailed command information and command examples about IPv4 and IPv6 software ACLs as applied to Routing and Multicasting. See all relevant Routing commands and configurations in “Layer Three, Switching and Routing” and all relevant Multicast commands and configurations in “Multicast Applications”.

Defining Hardware MAC ACLs

These are used to filter traffic based on specific source or destination MAC addresses contained within the data frames. They can be applied to ports in the form of access groups.

A MAC access list requires the following components:

- an ACL number in the range 4000-4699
- an action, permit, deny etc. See “Actions for Hardware ACLs” on page 31.7
- a source MAC address. You can use the format, HHHH.HHHH.HHHH to filter on a specific MAC address (where H is a hexadecimal number), or you can filter on any source MAC address by entering the word "any".
- a source MAC mask. This mask determines which portion of the source MAC address header will be compared with that found in the incoming packets. The mask is configured in the format <HHHH.HHHH.HHHH> where each H is a hexadecimal number. In practice each hex number will normally be either 0 (to represent a match) or F (to represent a don’t care condition). A mask is not required if the source address is specified as "any".
- a destination MAC address. You can use the format, HHHH.HHHH.HHHH to filter on a specific MAC address (where H is a hexadecimal number), or you can filter on any destination MAC address by entering the word "any".
- a destination MAC mask. This mask determines which portion of the destination MAC address header will be compared with that found in the incoming packets. The mask is configured in the format <HHHH.HHHH.HHHH> where each H is a hexadecimal number. In practice each hex number will normally be either 0 (to represent a match) or F (to represent a don’t care condition). A mask is not required if the source address is specified as "any".

Example To permit packets coming from a specific MAC address of 0030.841A.1234 and with any destination address:

```
         awplus# configure terminal
 awplus(config)# access-list 4000 permit 0030.841A.1234
                 0000.0000.0000 any
```

Defining Hardware IP ACLs

These are used to filter traffic based on specific source or destination IP addresses contained within the data frames. They can be applied to ports in the form of access groups.

An IP access list requires the following components:

- an ACL number in the range 3000-3699
- an action, see "Actions for Hardware ACLs" on page 31.7
- a packet type:
  - IP: This matches any type of IP packet. A source and destination address must also be specified, although they can be "any".
  - ICMP: This matches ICMP packets. A source and destination address must also be specified, although they can be "any". An ICMP type can optionally be specified after the destination address.
  - TCP: This matches TCP packets. A source and destination address must also be specified, although they can be "any". After the source address, a source port can optionally be specified and after the destination address a destination port can optionally be specified. The port matching can be done using **eq** (equal to), **gt** (greater than), **lt** (less than), **ne** (not equal to), or **range** (for a range of ports, which requires a start port and an end port).
  - UDP: This matches UDP packets and has the same options as TCP.
  - proto: This allows any IP protocol type to be specified. A source and destination address must be also specified, although they can be "any".

For example, to match (and permit) any type of IP packet containing a destination address of 192.168.1.1

```
awplus(config)# access-list 3000 permit ip any 192.168.1.1/32
```

To match (and permit) an ICMP packet with a source address of 192.168.x.x and an ICMP code of 4

```
awplus(config)# access-list 3001 permit icmp 192.168.0.0/16
                any icmp-type 4
```

To match a TCP packet with a source address of 192.168.x.x, source port of 80 and a destination port from 100 to 150:

```
awplus(config)# access-list 3002 permit tcp 192.168.0.0/16 eq
                80 any range 100 150
```

To match a UDP packet with a source address of 192.168.x.x, a destination address of 192.168.1.x, and a destination port greater than 80:

```
awplus(config)# access-list 3003 permit udp 192.168.0.0/16
                192.168.1.0/24 gt 80
```

Note that an IP address mask can be specified using either of the following notations:

- "A.B.C.D/M": This is the most common; e.g. 192.168.1.0/24
- "A.B.C.D A.B.C.D": 192.168.1.1 0.0.0.0 is the same as 192.168.1.1/32 and 192.168.1.1 255.255.255.255 is the same as "any"
- "host A.B.C.D": This is the same as A.B.C.D/32

Actions for Hardware ACLs

The following actions are available for Hardware ACLs:

deny:	Discard the packet.
permit:	Allow the packet.
copy-to-cpu:	Send a copy of the packet to the CPU and forward it as well. This is the same as copy,forward in AW hardware filters.
send-to-cpu:	Send the packet to the CPU and do not forward it. This is the same as copy, discard in AlliedWare hardware filters.
send-to-mirror:	Send the packet to the mirror port so packets are not switched
copy-to-mirror:	Send a copy of the packet to the mirror port and forward it as well.

Attaching hardware ACLs to interfaces

A hardware ACL is attached directly to a switchport using the access-group command. For example, to permit traffic from 192.168.1.x, but discard from 192.168.x.x:

```
        awplus# configure terminal

awplus(config)# access-list 3000 permit ip 192.168.1.0/24
                any

awplus(config)# access-list 3001 deny ip 192.168.0.0/24 any

awplus(config)# interface port1.0.1

awplus(config-if)# access-group 3000

awplus(config-if)# access-group 3001
```

Hardware ACLs and QoS classifications

Interface ACLs and QoS policies can both be attached to the same port. Where this is done, packets received on the port will be matched against the ACLs first.

The interface ACLs and QoS classifications are implemented by taking the first matching filter and applying the action defined for that filter. All subsequent matches in the table are then ignored. Thus, because ACLs are also matched first, if the matching ACL has a permit action, the packet is forwarded due to that rule's action and any subsequent QoS rules are bypassed.

You can also apply permit rules using QoS.

For example, you might want to permit a source IP address of 192.168.1.x, but block everything else on 192.168.x.x.

In this case you could create both the permit and deny rules using QoS.

Classifying Your Traffic

Classification is the process of **filtering** and **marking**. Filtering involves sorting your data into appropriate traffic types. Marking involves tagging the data so that downstream ports and routers can apply appropriate service policy rules.

There are two reasons to classify data:

1. To provide network security (Security ACLs)
2. To apply service quality criteria QoS.

Security ACLs

The main application of security ACLs is to block undesired traffic. Other applications include:

- copy-to-cpu
- copy-to-mirror
- send-to-cpu
- send-to-mirror

For more information on these applications see “Actions for Hardware ACLs” on page 31.7

QoS ACLs

When using ACLs though QoS, the same classification and action abilities are available, but QoS has some additional fields that it can match on (see Match Commands) and also provides the ability to perform metering, marking and remarking on packets that match the filter definitions.

The action used by a QoS class-map is determined by the ACL that is attached to it. If no ACL is attached, it uses the permit action. If an ACL is not required by the class-map (for example, only matching on the VLAN) and a deny action is required, a MAC ACL should be added with `any` for source address and `any` for destination address.

The following example creates a class-map with will deny all traffic on vlan 2:

```
     awplus(config)# access-list 4000 deny any any
     awplus(config)# class-map cmap1
awplus(config-cmap)# match access-group 4000
awplus(config-cmap)# match vlan 2
```

The default class-map matches to all traffic and so cannot have any match or ACL commands applied to it. The action for this class-map is set via the default-action command and is `permit` by default. It can be changed to `deny` by using the following commands:

```
     awplus(config)# policy-map pmap1
awplus(config-pmap)# default-action deny
```

For more information on applying QoS filtering, see “Classifying your Data” on page 35.7.

Attaching hardware ACLs using QoS

The same functionality can be achieved using QoS, by attaching the ACL to a class-map, attaching the class-map to a policy-map and attaching the policy-map to a port:

Step 1: Enable QoS on the switch

```
awplus(config)# mls qos enable
```

Step 2: Create access lists

Create ACL `3000` to permit all packets from the 192.168.1 subnet:

```
awplus(config)# access-list 3000 permit ip 192.168.1.0/24 any
```

Create ACL `3001` to deny all packets from the 192.168.0 subnet.:

```
awplus(config)# access-list 3001 deny ip 192.168.0.0/24 any
```

Step 3: Attach access-groups to class-maps

Attach ACL 3000 to the class-map cmap1:

```
      awplus(config)# class-map cmap1
 awplus(config-cmap)# match access-group 3000
 awplus(config-cmap)# exit
```

Attach ACL 3001 to the same class-map (cmap2):

```
 awplus(config-cmap)# match access-group 3001
 awplus(config-cmap)# exit
```

Step 4: Attach class-maps to policy-maps

Attach the class-map cmap1 to policy-map pmap1:

```
       awplus(config)# policy-map pmap1
  awplus(config-pmap)# class cmap1
awplus(config-pmap-c)# exit
```

Add the class-map cmap2 to the policy-map pmap1:

```
  awplus(config-pmap)# class cmap2
awplus(config-pmap-c)# exit
```

Return to Global Configuration mode:

```
  awplus(config-pmap)# exit
```

Step 5: Add policy-maps to ports

Add policy-map pmap1 to port1.0.1:

```
    awplus(config)#  interface port1.0.1
 awplus(config-if)#  service-policy input pmap1
```

Note that multiple interface ACLs can be attached to the same port, or either type and can be interleaved. The order of matching is based on the order in which the ACLs were attached to the port. Only one ACL can be attached to a class-map, but multiple class-maps can be attached to a policy-map. Interface ACLs can be attached to the same port as a QoS policy, with the interface ACLs being matched first as described at the beginning of the Classification section.

Filtering hardware ACLs with QoS

Another reason for using QoS rather than interface ACLs is that QoS provides a lot more fields on which to match. These are accessed through the match commands in config-cmap mode.

Config-cmap mode describes the fields that can be matched on. Only one of each type can be matched, with the exception of tcp-flags (see below for classification). If multiple matches are specified, they are ANDed together.

The following example shows how you can match a packet on vlan 2, that has a source IP address of 192.168.x.x and a DSCP of 12:

Create ACL 3000 to permit all packets from the 192.168 subnet.:

```
        awplus#  configure terminal
awplus(config)#  access-list 3000 permit ip 192.168.0.0/16 any
```

Apply ACL 3000 to the class-map cmap1 and add the matching criteria of vlan 2 and DSCP 12:

```
     awplus(config)#  class-map cmap1
awplus(config-cmap)#  match access-group 3000
awplus(config-cmap)#  match vlan 2
awplus(config-cmap)#  match dscp 12
awplus(config-cmap)#  exit
```

Using QoS Match Commands with TCP Flags

Usually, if multiple matches of the same type are specified, the matching process will apply to the last match that you specified. For TCP flags however, the arguments are ANDed together. For example, the following series of commands will match on a packet that has ack, syn and fin set:

```
         awplus# configure terminal
 awplus(config)# class-map cmap1
awplus(config-cmap)# match tcp-flags ack
awplus(config-cmap)# match tcp-flags syn
awplus(config-cmap)# match tcp-flags fin
awplus(config-cmap)# exit
```

The following commands will achieve the same result:

```
         awplus# configure terminal
 awplus(config)# class-map cmap1
awplus(config-cmap)# match tcp-flags ack syn fin
awplus(config-cmap)# exit
```

Note that the matching is looking to see whether "any" of the specified flags are set. There is no checking for whether any of these flags are unset. Therefore the following commands will match on a packet in any of the following combinations of syn and ack status flags as shown in the following table:

```
         awplus# configure terminal
 awplus(config)# class-map cmap1
awplus(config-cmap)# match tcp-flags syn
awplus(config-cmap)# exit
```

Syn	Ack	Match on Packet
Set	Set	Yes
Set	Unset	Yes
Unset	Set	No
Unset	Unset	No

If you want to drop packets with syn only, but not with ack and syn, the following two class-maps can be used (note that ACL 4000 is used to apply a drop action as described in "Actions for Hardware ACLs" on page 31.7):

Step 1: Create access lists

Create ACL 4000 to deny all packets with any source or destination address:

```
          awplus# configure terminal
  awplus(config)# access-list 4000 deny any any
```

Step 2: Create class-maps

Create the class-map cmap1 and configure it to match on the TCP flags, ack and syn:

```
     awplus(config)# class-map cmap1
awplus(config-cmap)# match tcp-flags ack syn
awplus(config-cmap)# exit
```

Create the class-map cmap2 and configure it to match on the TCP flag, syn:

```
     awplus(config)# class-map cmap2
awplus(config-cmap)# match tcp-flags syn
```

Step 3: Apply access-groups to class-maps

Apply ACL 4000 to this class-map (i.e. to cmap2):

```
awplus(config-cmap)# match access-group 4000
awplus(config-cmap)# exit
```

Step 4: Create policy-maps

Create the policy-map pmap1 and associate it with cmap1:

```
       awplus(config)# policy-map pmap1
  awplus(config-pmap)# class cmap1
awplus(config-pmap-c)# exit
```

Step 5: Associate class-maps with policy-maps

Associate cmap2 with this policy-map (pmap1):

```
  awplus(config-pmap)# class cmap2
awplus(config-pmap-c)# exit
```

ACL Filter Sequence Numbers

To help you manage ACLs you can apply sequence numbers to filters. This allows you to remove filters from named and numbered ACLs without having to reconfigure an ACL.

The ability to add sequence numbers to filters simplifies updates through the ability to position a filter within an ACL. When you add a new filter, you can specify a sequence number to position the filter in the ACL and you can also remove a current filter in an ACL by specifying a sequence number.

ACL Filter Sequence Number Behavior

- If filters with no sequence numbers are applied then the first filter is assigned a sequence number of 10, and successive filters are incremented by 10. Sequence numbers are generated automatically if they are not specified at entry.
- The maximum filter sequence number is 65535. If the sequence number exceeds this maximum, the command will not be recognized and will show the error message:
  `% Unrecognized command`
- If you enter a filter without a sequence number it is assigned a sequence number that is 10 greater than the last sequence number and is placed at the end of the ACL.
- If you enter a filter that matches an already existing filter then the first filter is overwritten with the subsequent filter.
- ACL sequence numbers determine the order of execution of filters in an ACL. Filters in a ACL with a lower value sequence number are executed before filters with a higher value.
- Output from show running-config displays ACL entries without filter sequence numbers. Output from relevant **show** commands displays ACL entries with their sequence numbers.
- ACL sequence numbers are re-numbered upon switch restart following a reload command, or after powering off and powering on the switch. ACL sequence numbers are renumbered starting from 10 and increment by 10 for each filter. See the sample output in the configuration section that follows for an illustration of this behavior. No ACL sequence number re-number command is available to perform this action.
- The ACL sequence number feature works with numbered and named standard and extended IPv4 and IPv6 access lists, plus named hardware IPv4 and IPv6 access lists
- The name of an access list can be designated as a number. Number in named ACLs must not exist within the range or designated numbered ACLs. (where <1-99> and <1300-1999> are standard numbered ACLs, <100-199> and <2000-2699> are extended numbered ACLs, <3000-3699> and <4000-4699> are hardware numbered ACLs).

ACL Filter Sequence Number Applicability

The ACL sequence number support feature is available with numbered and named standard and extended IPv4 and IPv6 ACLs, and the named hardware IPv4 and IPv6 ACLs.

Numbered standard ACLs are available in the range `<1-99>` and `<1300-1999>`, which permit or deny source addresses to control packets coming from network devices or hosts, in software.

Numbered extended ACLs are available in the range `<100-199>` and `<2000-2699>`, which permit or deny source addresses and destination addresses (plus ICMP, TCP, UDP messages) to control packets coming from and going to network devices or hosts.

Named hardware IPv4 and IPv6 ACLs are available which permit or deny IP and MAC source and destination addresses plus VLAN IDs to control packets coming from and going to network device and hosts. Named hardware IPv4 and IPv6 ACLs use the ACL sequence number support feature for ACL revision.

The ACL sequence number support feature is available for use with named hardware IPv4 and IPv6 ACLs, but this feature is not available for use with the numbered hardware IPv4 ACLs.

Numbered hardware ACLs are available in the range `<3000-3699>`, which permit or deny IP source addresses, IP destination addresses, and VLAN IDs to control packets coming from and going to network devices and hosts, in hardware.

Numbered hardware ACLs are available in the range `<4000-4699>`, which permit or deny MAC source addresses, MAC destination addresses, and VLAN IDs to control packets coming from and going to network devices and hosts, in hardware.

ACL Filter Sequence Number Types

There are ACL filter sequence numbers available for the following types of ACLs:

ACL Type	ACL Command Syntax
IPv4 Standard Numbered ACLs	**access-list <1-99>** **access-list <1300-1999>**
IPv4 Extended Numbered ACLs	**access-list <100-199>** **access-list <2000-2699>**
IPv4 Standard Named ACLs	**access-list standard <*name*>**
IPv4 Extended Named ACLs	**access-list extended <*name*>**
IPv4 Hardware Named ACLs	**access-list hardware <*name*>**
IPv6 Standard Named ACLs	**ipv6 access-list standard <*name*>**
IPv6 Extended Named ACLs	**ipv6 access-list extended <*name*>**
IPv6 Hardware Named ACLs	**ipv6 access-list <*name*>**

Note that ACL sequence number support for these ACL commands is optional not required. An ACL sequence number will be added automatically, starting at 10 and incrementing by 10.

ACL Commands Without ACL Filter Sequence Numbers

ACL filter sequence numbers is not available for numbered hardware ACL commands:

access-list <3000-3699>
access-list <4000-4699>

ACL Filter Sequence Number Entry Examples

See the below CLI entry examples for prompt sub-modes for ACL filters after ACL commands:

- To create an IPv4 Standard ACL and then define ACL filters at the IPv4 Standard ACL Configuration mode prompt **awplus(config-ip-std-acl)#**, enter the following commands:

```
          awplus(config)# access-list 1

awplus(config-ip-std-acl)# permit 192.168.1.0 0.0.0.255


          awplus(config)# access-list standard std_name

awplus(config-ip-std-acl)# permit 192.168.1.0/24
```

- To create an IPv4 Extended ACL and then define ACL filters at the IPv4 Extended ACL Configuration mode prompt **awplus(config-ip-ext-acl)#**, enter the following commands:

```
          awplus(config)# access-list 100

awplus(config-ip-ext-acl)# permit ip 192.168.1.0 0.0.0.255
                           192.168.2.0 0.0.0.255


          awplus(config)# access-list extended ext_name

awplus(config-ip-ext-acl)# permit ip 192.168.1.0 0.0.0.255
                           192.168.2.0 0.0.0.255
```

- To create an IPv4 Hardware ACL and then define ACL filters at the IPv4 Hardware ACL Configuration mode prompt **awplus(config-ip-hw-acl)#**, enter the following commands:

```
         awplus(config)# access-list hardware hw_name

awplus(config-ip-hw-acl)# permit ip 192.168.1.0 0.0.0.255
                          192.168.2.0 0.0.0.255
```

- To create an IPv6 Standard ACL and then define ACL filters at the IPv6 Standard ACL Configuration mode prompt **awplus(config-ipv6-std-acl)#**, enter the following commands:

```
            awplus(config)# ipv6 access-list standard
                            ipv6_std_name

awplus(config-ipv6-std-acl)# permit 2001:db8::/64
```

- To create an IPv6 Extended ACL and then define ACL filters at the IPv6 Extended

Configuration mode prompt **awplus(config-ipv6-ext-acl)#**, enter the following commands:

```
           awplus(config)# ipv6 access-list extended
                           ipv6_ext_name

  awplus(config-ipv6-ext-acl)# permit ip 2001:db8::/64
                           2001:db9::/64
```

ACL Filter Sequence Configuration

First create a named or numbered ACL to enter ACL filters in the ACL sub-modes available:

Step 1: Create a new ACL and add a new filter

Create ACL 10 and then add a new filter to the access-list to permit all packets from the 192.168.1 subnet:

```
                awplus# configure terminal

        awplus(config)# access-list 10

awplus(config-ip-std-acl)# permit 192.168.1.0 0.0.0.255

awplus(config-ip-std-acl)# end

                awplus# show access-list 10
```

```
Standard IP access list 10
   10 permit 192.168.1.0, wildcard bits 0.0.0.255
```

Step 2: Add another filter to the ACL

Append to, or add at the end of, ACL 10 a new filter to deny all packets from the 192.168.2 subnet:

```
                awplus# configure terminal

        awplus(config)# access-list 10

awplus(config-ip-std-acl)# deny 192.168.2.0 0.0.0.255

awplus(config-ip-std-acl)# end

                awplus# show access-list 10
```

```
Standard IP access list 10
   10 permit 192.168.1.0, wildcard bits 0.0.0.255
   20 deny   192.168.2.0, wildcard bits 0.0.0.255
```

Note that if you add a filter to an ACL without specifying a sequence number the new filter is automatically assigned a sequence number. Sequence numbers are assigned in multiples of ten from the sequence number of the last filter.

Step 3: Insert a filter into the ACL

Insert a new filter with the sequence number 15 into ACL 10 to permit packets from the 192.168.3 subnet:

```
                awplus# configure terminal
        awplus(config)# access-list 10
awplus(config-ip-std-acl)# 15 permit 192.168.3.0 0.0.0.255
awplus(config-ip-std-acl)# end
                awplus# show access-list 10
```

```
Standard IP access list 10
   10 permit 192.168.1.0, wildcard bits 0.0.0.255
   15 permit 192.168.3.0, wildcard bits 0.0.0.255
   20 deny   192.168.2.0, wildcard bits 0.0.0.255
```

The new filter has precedence over the filter with the sequence number 20.

Step 4: Remove a filter from the ACL by specifying a filter pattern

Remove the filter with the IP address 192.168.2 from ACL 10:

```
                awplus# configure terminal
        awplus(config)# access-list 10
awplus(config-ip-std-acl)# no deny 192.168.2.0 0.0.0.255
awplus(config-ip-std-acl)# end
                awplus# show access-list 10
```

```
Standard IP access list 10
   10 permit 192.168.1.0, wildcard bits 0.0.0.255
   15 permit 192.168.3.0, wildcard bits 0.0.0.255
```

Step 5: Remove a filter from the ACL by specifying a sequence number

Remove the filter with the sequence number 10 from ACL 10:

```
               awplus# configure terminal
       awplus(config)# access-list 10
awplus(config-ip-std-acl)# no 10
awplus(config-ip-std-acl)# end
               awplus# show access-list
```

```
Standard IP access list 10
   15 permit 192.168.3.0, wildcard bits 0.0.0.255
```

Creating ACLs in Global Configuration Mode

You can add new filters in **Global Configuration** mode with the access-list extended (named) command on page 33.4. In this mode the filters are assigned a sequence number corresponding to the order in which there are entered, i.e. the first filter entered has higher precedence in the ACL.

Step 1: Add filters with the **access-list** command

Add filters to ACL 10 using the **access-list** command:

```
       awplus# configure terminal
awplus(config)# access-list 10 permit 192.168.1.0 0.0.0.255
awplus(config)# access-list 10 deny 192.168.2.0 0.0.0.255
awplus(config)# end
       awplus# show access-list 10
```

```
Standard IP access list 10
   15 permit 192.168.3.0, wildcard bits 0.0.0.255
   20 permit 192.168.1.0, wildcard bits 0.0.0.255
   30 deny 192.168.2.0, wildcard bits 0.0.0.255
```

You can then enter the **IPv4 Standard ACL Configuration** mode and use the (access-list standard named filter) command on page 33.32 to specify sequence numbers to reorder the filters.

Step 2: Reorder the filters

Reorder the filters in ACL 10 by specifying a sequence number for each filter. The specified sequence number will overwrite the previous sequence number assigned to the filter:

```
                  awplus# configure terminal
          awplus(config)# access-list 10
awplus(config-ip-std-acl)# 1021 permit 192.168.1.0 0.0.0.255
awplus(config-ip-std-acl)# 3333 permit 192.168.3.0 0.0.0.255
awplus(config-ip-std-acl)# 2772 deny 192.168.2.0 0.0.0.255
awplus(config-ip-std-acl)# end
                  awplus# show access-list 10
```

```
Standard IP access list 10
 1021 permit 192.168.1.0, wildcard bits 0.0.0.255
 2772 deny   192.168.2.0, wildcard bits 0.0.0.255
 3333 permit 192.168.3.0, wildcard bits 0.0.0.255
```

Step 3: Copy the running-config file into the startup-config file

Copy the running-config into the file set as the current startup-config file and then reload the device. Before the reload occurs, you will receive a confirmation request saying: “`reboot system? (y/n):`”.

After the device has reboot the sequence numbers of the filters in the ACL have been reassigned incrementing from 10.

Display the ACL configuration details

Display the running system status and configuration details for ACLs:

```
awplus# show running-config access-list
```

```
!
access-list 1 deny 10.1.1.0 0.0.0.255
access-list 1 permit any
access-list 2
access-list 5
access-list 10 permit 192.168.1.0 0.0.0.255
access-list 10 deny 192.168.2.0 0.0.0.255
access-list 10 permit 192.168.3.0 0.0.0.255
access-list 20
access-list 25 permit 10.1.2.0 0.0.0.255
access-list 25 deny 192.168.1.0 0.0.0.255
access-list 50
access-list 95 permit any
access-list 100
access-list 1300
access-list 2000
access-list extended acl
access-list extended my-list
access-list extended name
access-list extended name1
access-list standard name3
ipv6 access-list extended ipv6_acl
ipv6 access-list standard ipv6_acl2
ipv6 access-list extended my-ipv6-list
ipv6 access-list extended my-list
ipv6 access-list standard my-new-list
ipv6 access-list standard name
ipv6 access-list standard name1 deny any
ipv6 access-list extended name5
ipv6 access-list standard name6
access-list hw_acl
access-list icmp
access-list my-hw-list
access-list name2
access-list name4
!
```

For more information see show running-config access-list command on page 7.40.

Chapter 32: IPv4 Hardware Access Control List (ACL) Commands

Introduction

This chapter provides an alphabetical reference for the IPv4 Hardware Access Control List (ACL) commands, and contains detailed command information and command examples about IPv4 hardware ACLs, which are applied directly to interfaces using the access-group command.

Note See Chapter 31, Access Control Lists Introduction for descriptions of ACLs, and for further information about rules when applying ACLs see the ACL Rules section.

See ACL Filter Sequence Numbers and ACL Filter Sequence Number Behavior sections in Chapter 31, Access Control Lists Introduction about ACL Filters.

To apply ACLs to an LACP channel group, apply it to all the individual switch ports in the channel group. To apply ACLs to a static channel group, apply it to the static channel group itself. For more information on link aggregation see Chapter 20, Link Aggregation Introduction and Configuration, and Chapter 21, Link Aggregation Commands.

Note Text in parenthesis in command names indicates usage not keyword entry. For example, **access-list hardware (named)** indicates named IPv4 hardware ACLs entered as `access-list hardware <name>` where `<name>` is a placeholder not a keyword.

Note Parenthesis surrounding ACL filters indicates the type of ACL filter not the keyword entry in the CLI, such as **(access-list standard numbered filter)** represents command entry in the format shown in the syntax `[<sequence-number>] {deny|permit} {<source>|host <host-address>|any}`.

Note Software ACLs will **deny** access unless **explicitly permitted** by an ACL action.

IPv4 Hardware Access List Commands and Prompts

Many of the ACL commands operate from sub-modes that are specific to particular ACL types. The table "IPv4 Hardware Access List Commands and Prompts" shows the CLI prompts at which ACL commands are entered.

Table 32-1: IPv4 Hardware Access List Commands and Prompts

Command Name	Command Mode	Prompt
show interface access-group	Privileged Exec	`awplus#`
show access-list (IPv4 Hardware ACLs)	Privileged Exec	`awplus#`
show interface access-group	Privileged Exec	`awplus#`
access-group	Global Configuration	`awplus(config)#`
access-list (hardware IP numbered)	Global Configuration	`awplus(config)#`
access-list (hardware MAC numbered)	Global Configuration	`awplus(config)#`
access-list hardware (named)	Global Configuration	`awplus(config)#`
access-group	Interface Configuration	`awplus(config-if)#`
(access-list hardware ICMP filter)	IPv4 Hardware ACL Configuration	`awplus(config-ip-hw-acl)#`
(access-list hardware IP protocol filter)	IPv4 Hardware ACL Configuration	`awplus(config-ip-hw-acl)#`
(access-list hardware MAC filter)	IPv4 Hardware ACL Configuration	`awplus(config-ip-hw-acl)#`
(access-list hardware TCP UDP filter)	IPv4 Hardware ACL Configuration	`awplus(config-ip-hw-acl)#`
commit (IPv4)	IPv4 Hardware ACL Configuration	`awplus(config-ip-hw-acl)#`

Command List

access-group

This command adds or removes a hardware-based access-list to a switch port interface. The number of hardware numbered and named access-lists that can be added to a switch port interface is determined by the available memory in hardware-based packet classification tables. This command works in both Global Configuration and Interface Configuration modes to apply hardware access-lists to all switch port interfaces or selected switch port interfaces respectively.

The **no** variant of this command removes the selected access-list from an interface.

Syntax

```
access-group [<3000-3699>|<4000-4699>|<hardware-access-list-name>]

no access-group [<3000-3699>|4000-4699|<hardware-access-list-name>]
```

Parameter	Description
<3000-3699>	Hardware IP access-list.
<4000-4699>	Hardware MAC access-list.
<hardware-access-list-name>	The hardware access-list name.

Mode Interface Configuration or Global Configuration

Default Any traffic on an interface controlled by a hardware ACL that does not explicitly match a filter is permitted.

Usage First create an IP access-list that applies the appropriate permit, deny requirements etc with the access-list (hardware IP numbered) command on page 32.6, the access-list (hardware MAC numbered) command on page 32.15 or the access-list hardware (named) command on page 32.17. Then use this command to apply this hardware access-list to a specific port or port range. Note that this command will apply the access-list only to incoming data packets.

To apply ACLs to an LACP aggregated link, apply it to all the individual switch ports in the aggregated group. To apply ACLs to a static channel group, apply it to the static channel group itself. Do not apply an ACL to a dynamic (LACP) or static aggregated link that spans more than one switch instance (Chapter 21, Link Aggregation Commands).

Note that you cannot apply software standard and extended numbered ACLs to switch port interfaces with the access-group command. This command will only apply hardware ACLs.

Note Hardware ACLs will **permit** access unless **explicitly denied** by an ACL action.

Examples To add the numbered hardware access-list 3005 to all switch ports, enter the following commands:

```
        awplus# configure terminal
awplus(config)# access-group 3005
```

To add the numbered hardware access-list 3005 to switch port interface port1.0.1, enter the following commands:

```
         awplus# configure terminal
 awplus(config)# interface port1.0.1
awplus(config-if)# access-group 3005
```

To add the named hardware access-list hw-acl to switch port interface port1.0.2, enter the following commands:

```
         awplus# configure terminal
 awplus(config)# interface port1.0.2
awplus(config-if)# access-group hw-acl
```

To apply an ACL to static channel group 2 containing switch port1.0.5 and port1.0.6, use the commands:

```
         awplus# configure terminal
 awplus(config)# interface port1.0.5-1.0.6
awplus(config-if)# static-channel-group 2
 awplus(config)# interface sa2
awplus(config-if)# access-group 3000
```

Related Commands

access-list hardware (named)
access-list (hardware IP numbered)
access-list (hardware MAC numbered)
show interface access-group

access-list (hardware IP numbered)

This command creates an access-list for use with hardware classification, such as QoS. The access-list will match on either TCP or UDP type packets that have the specified source and destination IP addresses and Layer 4 port values or ranges. The parameter **any** may be specified if an address does not matter and the port values are optional.

The **no** variant of this command removes the previously specified IP hardware access-list.

Syntax [ip]

```
access-list <3000-3699>
    {deny|permit|copy-to-cpu|copy-to-mirror|send-to-mirror|send-to-
    cpu} ip <source> <destination>
```

Syntax [icmp]

```
access-list <3000-3699>
    {deny|permit|copy-to-cpu|copy-to-mirror|send-to-mirror|send-to-
    cpu} icmp <source> <destination> [icmp-type <type-number>]

no access-list <3000-3699>
```

Table 32-2: Parameters in the access-list (hardware IP numbered) command - ip|icmp

Parameter	Description	
<3000-3699>	Hardware IP access-list number.	
deny	Access-list rejects packets that match the source and destination filtering specified with this command.	
permit	Access-list permits packets that match the source and destination filtering specified with this command.	
copy-to-cpu	Specify packets to copy to the CPU.	
copy-to-mirror	Specify packets to copy to the mirror port.	
send-to-mirror	Specify packets to send to the mirror port.	
send-to-cpu	Specify packets to send to the CPU.	
icmp	ICMP packet.	
ip	IP packet.	
<source>	The source address of the packets. You can specify a single host, a subnet, or all sources. The following are the valid formats for specifying the source:	
	any	Matches any source IP address.
	host *<ip-addr>*	Matches a single source host with the IP address given by *<ip-addr>* in dotted decimal notation.
	<ip-addr>/<prefix>	An IPv4 address, followed by a forward slash, then the prefix length. This matches any source IP address within the specified subnet.
	<ip-addr> *<reverse-mask>*	Alternatively, you can enter a reverse mask in dotted decimal format. For example, entering `192.168.1.1 0.0.0.255` is the same as entering `192.168.1.1/24`.

Table 32-2: Parameters in the access-list (hardware IP numbered) command - ip|icmp(cont.)

Parameter(cont.)	**Description**(cont.)	
<destination>	The destination address of the packets. You can specify a single host, a subnet, or all destinations. The following are the valid formats for specifying the destination:	
	`any`	Matches any destination IP address.
	`host` *<ip-addr>*	Matches a single destination host with the IP address given by *<ip-addr>* in dotted decimal notation.
	<ip-addr>/ *<prefix>*	An IPv4 address, followed by a forward slash, then the prefix length. This matches any destination IP address within the specified subnet.
	<ip-addr> *<reverse-mask>*	Alternatively, you can enter a reverse mask in dotted decimal format. For example, entering `192.168.1.1 0.0.0.255` is the same as entering `192.168.1.1/24`.
`icmp-type`	Matches only a specified type of ICMP messages. This is valid only when the filtering is set to match ICMP packets.	
<type-number>	The ICMP type, as defined in RFC792 and RFC950. Specify one of the following integers to create a filter for the ICMP message type:	
	0	Echo replies.
	3	Destination unreachable messages.
	4	Source quench messages.
	5	Redirect (change route) messages.
	8	Echo requests.
	11	Time exceeded messages.
	12	Parameter problem messages.
	13	Timestamp requests.
	14	Timestamp replies.
	15	Information requests.
	16	Information replies.
	17	Address mask requests.
	18	Address mask replies.

Syntax [tcp|udp]

```
access-list <3000-3699>
    {copy-to-cpu|copy-to-mirror|send-to-mirror|deny|permit|send-to-
    cpu} {tcp|udp} <source>
    {eq <sourceport>|lt <sourceport>|gt <sourceport>|ne <sourceport>|
    [range <start-range> <end-range>}
    <destination>
    [eq <destport>|lt <destport>|gt <destport>|ne <destport>]
    [range <start-range> <end-range>]
```

```
no access-list <3000-3699>
```

Table 32-3: Parameters in the access-list (hardware IP numbered) command - tcp|udp

<table>
<tr><th>Parameter</th><th colspan="2">Description</th></tr>
<tr><td><3000-3699></td><td colspan="2">Hardware IP access-list.</td></tr>
<tr><td>copy-to-cpu</td><td colspan="2">Specify packets to copy to the CPU.</td></tr>
<tr><td>copy-to-mirror</td><td colspan="2">Specify packets to copy to the mirror port.</td></tr>
<tr><td>send-to-mirror</td><td colspan="2">Specify packets to send to the mirror port.</td></tr>
<tr><td>deny</td><td colspan="2">The access-list rejects packets that match the type, source, and destination filtering specified with this command.</td></tr>
<tr><td>permit</td><td colspan="2">The access-list permits packets that match the type, source, and destination filtering specified with this command.</td></tr>
<tr><td>send-to-cpu</td><td colspan="2">Specify packets to send to the CPU.</td></tr>
<tr><td>tcp</td><td colspan="2">The access-list matches only TCP packets.</td></tr>
<tr><td>udp</td><td colspan="2">The access-list matches only UDP packets.</td></tr>
<tr><td rowspan="5"><source></td><td colspan="2">The source address of the packets. You can specify a single host, a subnet, or all sources. The following are the valid formats for specifying the source:</td></tr>
<tr><td>any</td><td>Matches any source IP address.</td></tr>
<tr><td>host <ip-addr></td><td>Matches a single source host with the IP address given by <ip-addr> in dotted decimal notation.</td></tr>
<tr><td><ip-addr>/<prefix></td><td>An IPv4 address, followed by a forward slash, then the prefix length. This matches any source IP address within the specified subnet.</td></tr>
<tr><td><ip-addr>
<reverse-mask></td><td>Alternatively, you can enter a reverse mask in dotted decimal format. For example, entering 192.168.1.1 0.0.0.255 is the same as entering 192.168.1.1/24.</td></tr>
</table>

Table 32-3: Parameters in the access-list (hardware IP numbered) command - tcp|udp(cont.)

Parameter(cont.)	**Description**(cont.)	
<destination>	The destination address of the packets. You can specify a single host, a subnet, or all destinations. The following are the valid formats for specifying the destination:	
	any	Matches any destination IP address.
	host *<ip-addr>*	Matches a single destination host with the IP address given by *<ip-addr>* in dotted decimal notation.
	<ip-addr>/<prefix>	An IPv4 address, followed by a forward slash, then the prefix length. This matches any destination IP address within the specified subnet.
	<ip-addr> *<reverse-mask>*	Alternatively, you can enter a reverse mask in dotted decimal format. For example, entering 192.168.1.1 0.0.0.255 is the same as entering 192.168.1.1/24.
<sourceport>	The source (TCP or UDP) port number, specified as an integer between 0 and 65535.	
range	Range of port numbers.	
<start-range>	Port number at start of range *<0-65535>*.	
<end-range>	Port number at end of range *<0-65535>*.	
<destport>	The destination (TCP or UDP) port number, specified as an integer between 0 and 65535.	
eq	Matches port numbers that are equal to the port number specified immediately after this parameter.	
lt	Matches port numbers that are less than the port number specified immediately after this parameter.	
gt	Matches port numbers that are greater than the port number specified immediately after this parameter.	
ne	Matches port numbers that are not equal to the port number specified immediately after this parameter.	
range	Range of port numbers.	
<start-range>	Port number at start of range *<0-65535>*.	
<end-range>	Port number at end of range *<0-65535>*.	

Syntax [proto]

```
access-list <3000-3699>
    {copy-to-cpu|copy-to-mirror|send-to-mirror|deny|permit|send-to-
    cpu} proto <ip-protocol> <source> <destination>

no access-list <3000-3699>
```

Table 32-4: Parameters in the access-list (hardware IP numbered) command - proto

Parameter	Description	
`<3000-3699>`	Hardware IP access-list.	
`copy-to-cpu`	Specify packets to copy to the CPU.	
`copy-to-mirror`	Specify packets to copy to the mirror port.	
`send-to-mirror`	Specify packets to send to the mirror port	
`deny`	Access-list rejects packets that match the source and destination filtering specified with this command.	
`permit`	Access-list permits packets that match the source and destination filtering specified with this command.	
`send-to-cpu`	Specify packets to send to the CPU.	
`<source>`	The source address of the packets. You can specify a single host, a subnet, or all sources. The following are the valid formats for specifying the source:	
	`any`	Matches any source IP address.
	`host <ip-addr>`	Matches a single source host with the IP address given by `<ip-addr>` in dotted decimal notation.
	`<ip-addr>/<prefix>`	An IPv4 address, followed by a forward slash, then the prefix length. This matches any source IP address within the specified subnet.
	`<ip-addr> <reverse-mask>`	Alternatively, you can enter a reverse mask in dotted decimal format. For example, entering `192.168.1.1 0.0.0.255` is the same as entering `192.168.1.1/24`.

Table 32-4: Parameters in the access-list (hardware IP numbered) command - proto

Parameter(cont.)	**Description**(cont.)	
`<destination>`	The destination address of the packets. You can specify a single host, a subnet, or all destinations. The following are the valid formats for specifying the destination:	
	`any`	Matches any destination IP address.
	`host <ip-addr>`	Matches a single destination host with the IP address given by `<ip-addr>` in dotted decimal notation.

	`<ip-addr>/`
	`<prefix>`
		An IPv4 address, followed by a forward slash, then the prefix length. This matches any destination IP address within the specified subnet.

	`<ip-addr>`
	`<reverse-mask>`
		Alternatively, you can enter a reverse mask in dotted decimal format. For example, entering `192.168.1.1 0.0.0.255` is the same as entering `192.168.1.1/24`.

`proto`	Matches only a specified type of IP Protocol <1-255>.	
`<ip-protocol>`	The IP protocol number, as defined by IANA (Internet Assigned Numbers Authority http://www.iana.org/assignments/protocol-numbers)	
	Protocol Number	Protocol Description [RFC Reference]
	1	Internet Control Message [RFC792]
	2	Internet Group Management [RFC1112]
	3	Gateway-to-Gateway [RFC823]
	4	IP in IP [RFC2003]
	5	Stream [RFC1190] [RFC1819]
	6	TCP (Transmission Control Protocol) [RFC793]
	8	EGP (Exterior Gateway Protocol) [RFC888]
	9	IGP (Interior Gateway Protocol) [IANA]
	11	Network Voice Protocol [RFC741]
	17	UDP (User Datagram Protocol) [RFC768]
	20	Host monitoring [RFC869]
	27	RDP (Reliable Data Protocol) [RFC908]
	28	IRTP (Internet Reliable Transaction Protocol) [RFC938]
	29	ISO-TP4 (ISO Transport Protocol Class 4) [RFC905]

Table 32-4: Parameters in the access-list (hardware IP numbered) command - proto

Parameter(cont.)	**Description**(cont.)	

`<ip-protocol>`
`(cont.)`
	30	Bulk Data Transfer Protocol [RFC969]

	33	DCCP (Datagram Congestion Control Protocol) [RFC4340]
	48	DSR (Dynamic Source Routing Protocol) [RFC4728]
	50	ESP (Encap Security Payload) [RFC2406]
	51	AH (Authentication Header) [RFC2402]
	54	NARP (NBMA Address Resolution Protocol) [RFC1735]
	58	ICMP for IPv6 [RFC1883]
	59	No Next Header for IPv6 [RFC1883]
	60	Destination Options for IPv6 [RFC1883]
	88	EIGRP (Enhanced Interior Gateway Routing Protocol)
	97	Ethernet-within-IP Encapsulation / RFC3378
	98	Encapsulation Header / RFC1241
	108	IP Payload Compression Protocol / RFC2393
	112	Virtual Router Redundancy Protocol / RFC3768
	134	RSVP-E2E-IGNORE / RFC3175
	135	Mobility Header / RFC3775
	136	UDPLite / RFC3828
	137	MPLS-in-IP / RFC4023
	138	MANET Protocols / RFC-ietf-manet-iana-07.txt
	139-252	Unassigned / IANA
	253	Use for experimentation and testing / RFC3692
	254	Use for experimentation and testing / RFC3692
	255	Reserved / IANA

Mode Global Configuration

Default Any traffic on an interface controlled by a hardware ACL that does not explicitly match a filter is permitted.

Usage This command creates an access-list for use with hardware classification, such as when applying QoS. This command can be used to match ICMP packets, IP protocols, or TCP/UDP packets.

For ICMP packets, the <3000-3699> range IP hardware access-list will match any ICMP packet that has the specified source and destination IP addresses and ICMP type.

You may apply the **any** parameter if the source or destination IP address is not important. The ICMP type is an optional parameter.

Note Hardware ACLs will **permit** access unless **explicitly denied** by an ACL action.

Examples Follow the below example commands to configure access-lists for ICMP, IP protocol and TCP.

ICMP Example To create an access-list that will permit ICMP packets with a source address of `192.168.1.0/24` with any destination address and an ICMP type of 5 enter the below commands:

```
        awplus# configure terminal
awplus(config)# access-list 3000 permit icmp 192.168.1.0/24
                any icmp-type 5
```

To destroy the access-list with an access-list identity of `3000` enter the below commands:

```
        awplus# configure terminal
awplus(config)# no access-list 3000
```

IP Example To create an access-list that will permit any type of IP packet with a source address of `192.168.1.1` and any destination address, enter the commands:

```
        awplus# configure terminal
awplus(config)# access-list 3000 permit ip 192.168.1.1/32 any
```

To create an access-list that will deny all IGMP packets (IP protocol 2) from the `192.168.0.0` network, enter the commands:

```
        awplus# configure terminal
awplus(config)# access-list 3000 deny proto 2 192.168.0.0/16 any
```

TCP Example To create an access-list that will permit TCP packets with a destination address of `192.168.1.1`, a destination port of 80 and any source address and source port, enter the commands:

```
        awplus# configure terminal
awplus(config)# access-list 3000 permit tcp any 192.168.1.1/32
                eq 80
```

copy-to-mirror Example

To create an access-list that will copy-to-mirror TCP packets with a destination address of 192.168.1.1, a destination port of 80 and any source address and source port for use with the mirror interface command, enter the commands:

```
awplus#          configure terminal

awplus(config)#  access-list 3000 copy-to-mirror tcp any
                 192.168.1.1/32 eq 80
```

Related Commands

access-group
mirror interface
show running-config
show access-list (IPv4 Hardware ACLs)

access-list (hardware MAC numbered)

This command creates an access-list for use with hardware classification, such as QOS. The access-list will match on packets that have the specified source and destination MAC addresses. The parameter **any** may be specified if an address does not matter.

The **no** variant of this command removes the specified MAC hardware filter access-list.

Syntax

```
access-list <4000-4699>
    {copy-to-cpu|copy-to-mirror|deny|permit|send-to-cpu}
    {<source-mac-address> <source-mac-mask>|any}
    {<destination-mac-address> <destination-mac-mask>|any}

no access-list <4000-4699>
```

Parameter	Description
<4000-4699>	Hardware MAC access-list.
`copy-to-cpu`	Specify packets to copy to the CPU.
`copy-to-mirror`	Specify packets to copy to the mirror port.
`deny`	Access-list rejects packets that match the source and destination filtering.
`permit`	Access-list permits packets that match the source and destination filtering.
`send-to-cpu`	Specify packets to send to the CPU.
<source-mac-address>	The source MAC address of the packets. Enter this in the format <HHHH.HHHH.HHHH> Where each *H* is a hexadecimal number that represents a 4 bit binary number.
<source-mac-mask	The mask that will be applied to the source MAC addresses. Enter this in the format <HHHH.HHHH.HHHH> Where each *H* is a hexadecimal number that represents a 4 bit binary number. For a mask, each value will be either `0` or `F`. Where Hex `FF` = Ignore, and Hex `00` = Match.
`any`	Any source MAC address.
<destination-mac-address>	The destination MAC address of the packets. Enter this in the format <HHHH.HHHH.HHHH> Where each H is a hexadecimal number that represents a 4 bit binary number.
<destination-mac-mask>	The mask that will be applied to the destination MAC addresses. Enter this in the format <HHHH.HHHH.HHHH> Where each H is a hexadecimal number that represents a 4 bit binary number. For a mask, each value will be either `0` or `F`. Where Hex `FF` = Ignore, and Hex `00` = Match.
`any`	Any destination MAC address.

Mode Global Configuration

Default Any traffic on an interface controlled by a hardware ACL that does not explicitly match a filter is permitted.

Usage

This command creates an access-list for use with hardware classification, such as when applying QoS. The <4000-4699> range MAC hardware access-list will match on packets that have the specified source and destination MAC addresses. You may apply the **any** parameter if the source or destination MAC host address is not important.

Note Hardware ACLs will **permit** access unless **explicitly denied** by an ACL action.

Examples

To create an access-list that will permit packets with a MAC address of `0000.00ab.1234` and any destination address enter the commands:

```
        awplus# configure terminal

awplus(config)# access-list 4000 permit 0000.00ab.1234
                0000.0000.0000 any
```

To create an access-list that will permit packets with an initial MAC address component of `0000.00ab` and any destination address, enter the commands:

```
        awplus# configure terminal

awplus(config)# access-list 4001 permit 0000.00ab.1234
                0000.0000.FFFF any
```

To create an access-list that will copy-to-mirror packets with an initial MAC address component of `0000.00ab` and any destination address for use with the mirror interface command, enter the commands:

```
        awplus# configure terminal

awplus(config)# access-list 4001 copy-to-mirror 0000.00ab.1234
                0000.0000.FFFF any
```

To destroy the access-list with an access-list identity of `4000` enter the commands:

```
        awplus# configure terminal

awplus(config)# no access-list 4000
```

Related Commands

access-group
mirror interface
show running-config
show access-list (IPv4 Hardware ACLs)

access-list hardware (named)

This command creates a named hardware access-list that can be applied to a switch port interface. ACL filters for a named hardware ACL are created in the IPv4 Hardware ACL Configuration mode.

The **no** variant of this command removes the specified named hardware ACL.

Syntax

```
access-list hardware <hardware-access-list-name>
no access-list hardware <hardware-access-list-name>
```

Parameter	Description
<hardware-access-list-name>	Specify the hardware ACL name to then define ACL filters for in the subsequent IPv4 Hardware ACL Configuration mode.

Mode Global Configuration

Default Any traffic on an interface controlled by a hardware ACL that does not explicitly match a filter is permitted.

Usage Use this command to name a hardware ACL and enter the IPv4 Hardware ACL Configuration mode. If the named hardware ACL doesn't exist, it will be created after entry. If the named hardware ACL does exist, then you can enter IPv4 Hardware ACL Configuration mode for that existing ACL.

Entering this command with the hardware ACL name moves you to the `(config-ip-hw-acl)` prompt for the IPv4 Hardware ACL Configuration mode so you can enter ACL filters with sequence numbers. From this prompt, configure the filters for the ACL. See Chapter 31, Access Control Lists Introduction for complete examples of configured sequenced numbered ACLs.

See also the table "IPv4 Hardware Access List Commands and Prompts" in this chapter. This table shows the relevant prompts at which ACL commands and ACL filters are entered for sequenced ACLs.

Note Hardware ACLs will **permit** access unless **explicitly denied** by an ACL action.

Example To create the hardware access-list named `ACL-1` and enter the IPv4 Hardware ACL Configuration mode to specify the ACL filter entry, use the commands:

```
           awplus#  configure terminal
   awplus(config)#  access-list hardware ACL-1
awplus(config-ip-hw-acl)#
```

To remove the hardware access-list named `ACL-1`, use the commands:

```
         awplus#  configure terminal
 awplus(config)#  no access-list hardware ACL-1
```

Related Commands

access-group
(access-list hardware ICMP filter)
(access-list hardware IP protocol filter)
(access-list hardware TCP UDP filter)
(access-list standard named filter)
show access-list (IPv4 Hardware ACLs)

(access-list hardware ICMP filter)

Use this ACL filter to add a new ICMP filter entry to the current hardware access-list. The filter will match on any ICMP packet that has the specified source and destination IP addresses and ICMP type. The parameter **any** may be specified if an address does not matter and the ICMP type is an optional parameter. If a sequence number is specified, the new filter is inserted at the specified location. Otherwise, the new filter is added at the end of the access-list.

The **no** variant of this command removes an ICMP filter entry from the current hardware access-list. You can specify the ICMP filter entry for removal by entering either its sequence number (e.g. `no 10`), or by entering its ICMP filter profile without specifying its sequence number.

Note that the sequence number can be found by running the command, the show access-list (IPv4 Hardware ACLs) command on page 32.35.

Syntax [icmp]

```
[<sequence-number>]
   {deny|permit|send-to-cpu|copy-to-cpu|copy-to-mirror}
   icmp <source> <destination>
   [icmp <icmp-value>]

no {deny|permit|send-to-cpu|copy-to-cpu|copy-to-mirror}
   icmp <source> <destination>
   [icmp <icmp-value>]

no <sequence-number>
```

Parameter	Description
<sequence-number>	<1-65535> The sequence number for the filter entry of the selected access control list.
`deny`	Access-list rejects packets that match the source and destination filtering specified with this command.
`permit`	Access-list permits packets that match the source and destination filtering specified with this command.
`send-to-cpu`	Specify packets to send to the CPU.
`copy-to-cpu`	Specify packets to copy to the CPU.
`copy-to-mirror`	Specify packets to copy to the mirror port.
`icmp`	ICMP packet type.

Parameter(cont.)	**Description**(cont.)	
<*source*>	The source address of the packets. You can specify a single host, a subnet, or all sources. The following are the valid formats for specifying the source:	
	<*ip-addr*>/ <*prefix*>	An IPv4 address, followed by a forward slash, then the prefix length. This matches any source IP address within the specified subnet.
	<*ip-addr*> <*reverse-mask*>	Alternatively, you can enter a reverse mask in dotted decimal format. For example, entering 192.168.1.1 0.0.0.255 is the same as entering 192.168.1.1/24.
	host <*ip-addr*>	Matches a single source host with the IP address given by <*ip-addr*> in dotted decimal notation.
	any	Matches any source IP address.
<*destination*>	The destination address of the packets. You can specify a single host, a subnet, or all destinations. The following are the valid formats for specifying the destination:	
	<*ip-addr*>/ <*prefix*>	An IPv4 address, followed by a forward slash, then the prefix length. This matches any destination IP address within the specified subnet.
	<*ip-addr*> <*reverse-mask*>	Alternatively, you can enter a reverse mask in dotted decimal format. For example, entering 192.168.1.1 0.0.0.255 is the same as entering 192.168.1.1/24.
	host <*ip-addr*>	Matches a single destination host with the IP address given by <*ip-addr*> in dotted decimal notation.
	any	Matches any destination IP address.
icmp-type	The ICMP type.	
<*icmp-value*>	The value of the ICMP type.	

Mode IPv4 Hardware ACL Configuration

Default Any traffic on an interface controlled by a hardware ACL that does not explicitly match a filter is permitted.

Usage First create a named hardware access-list that applies the appropriate permit, deny requirements etc. Then use the access-group command on page 32.4 to apply this access-list to a specific port or range. Note that this command will apply the access-list only to **incoming** data packets.

An ACL can be configured with multiple ACL filters using sequence numbers. If the sequence number is omitted, the next available multiple of 10 will be used as the sequence number for the new filter. A new ACL filter can be inserted into the middle of an existing list by specifying the appropriate sequence number.

Note You must reach the prompt **awplus(config-ip-hw-acl)#** by running the access-list hardware (named) command on page 32.17, and entering an appropriate access-list name.

Note Hardware ACLs will **permit** access unless **explicitly denied** by an ACL action.

Example To add an access-list filter entry with a sequence number of `100` to the access-list named `my-list` that will permit ICMP packets with a source address of `192.168.1.0/24`, any destination address and an icmp type of 5, use the commands:

```
                awplus# configure terminal
        awplus(config)# access-list hardware my-list
awplus(config-ip-hw-acl)# 100 permit icmp 192.168.1.0/24 any
                          icmp-type 5
```

To remove an access-list filter entry with a sequence number of `100` in the access-list named `my-list`, use the commands:

```
                awplus# configure terminal
        awplus(config)# access-list hardware my-list
awplus(config-ip-hw-acl)# no 100
```

Related Commands

access-list hardware (named)
show running-config
show access-list (IPv4 Hardware ACLs)

(access-list hardware IP protocol filter)

Use this ACL filter to add an IP protocol type filter entry to the current hardware access-list. The filter will match on any IP packet that has the specified source and destination IP addresses and IP protocol type, or has the optionally specified source and destination MAC addresses. The parameter **any** may be specified if an address does not matter. If a sequence number is specified, the new filter is inserted at the specified location. Otherwise, the new filter is added at the end of the access-list.

The **no** variant of this command removes an IP protocol type filter entry from the current hardware access-list. You can specify the IP protocol type filter entry for removal by entering either its sequence number (e.g. `no 10`), or by entering its IP protocol type filter profile without specifying its sequence number.

Note that the sequence number can be found by running the show access-list (IPv4 Hardware ACLs) command on page 32.35.

Syntax [ip|proto]

```
[<sequence-number>]
   {deny|permit|send-to-cpu|copy-to-cpu|copy-to-mirror}
   {ip|any|proto <ip-protocol>}
   {<source>|dhcpsnooping} <destination>
   [mac {<mac-source-address> <mac-source-mask>|any}
   {<mac-destination-address> <mac-destination-mask>|any}]

no {deny|permit|send-to-cpu|copy-to-cpu|copy-to-mirror}
   {ip|any|proto <ip-protocol>}
   {<source>|dhcpsnooping} <destination>
   [mac {<mac-source-address> <mac-source-mask>|any}
   {<mac-destination-address> <mac-destination-mask>|any}]

no <sequence-number>
```

Parameter	Description
`<sequence-number>`	`<1-65535>` The sequence number for the filter entry of the selected access control list.
`deny`	Access-list rejects packets of the type specified.
`permit`	Access-list allows packets of the type specified
`send to cpu`	Specify packets to send to the CPU.
`copy to cpu`	Specify packets to copy to the CPU.
`copy to mirror`	Specify packets to copy to the mirror port.
`ip`	IP packets.
`any`	Any packet.
`proto <ip-protocol>`	The IP Protocol type specified by it protocol number <1-255>.

Parameter(cont.)	**Description**(cont.)
`<ip-protocol>`	The IP protocol number, as defined by IANA (Internet Assigned Numbers Authority http://www.iana.org/assignments/protocol-numbers)

Protocol Number	Protocol Description [RFC Reference]
1	Internet Control Message [RFC792]
2	Internet Group Management [RFC1112]
3	Gateway-to-Gateway [RFC823]
4	IP in IP [RFC2003]
5	Stream [RFC1190] [RFC1819]
6	TCP (Transmission Control Protocol) [RFC793]
8	EGP (Exterior Gateway Protocol) [RFC888]
9	IGP (Interior Gateway Protocol) [IANA]
11	Network Voice Protocol [RFC741]
17	UDP (User Datagram Protocol) [RFC768]
20	Host monitoring [RFC869]
27	RDP (Reliable Data Protocol) [RFC908]
28	IRTP (Internet Reliable Transaction Protocol) [RFC938]
29	ISO-TP4 (ISO Transport Protocol Class 4) [RFC905]
30	Bulk Data Transfer Protocol [RFC969]
33	DCCP (Datagram Congestion Control Protocol) [RFC4340]
48	DSR (Dynamic Source Routing Protocol) [RFC4728]
50	ESP (Encap Security Payload) [RFC2406]
51	AH (Authentication Header) [RFC2402]

Parameter(cont.)	**Description**(cont.)	
<ip-protocol> (cont.)	54	NARP (NBMA Address Resolution Protocol) [RFC1735]
	58	ICMP for IPv6 [RFC1883]
	59	No Next Header for IPv6 [RFC1883]
	60	Destination Options for IPv6 [RFC1883]
	88	EIGRP (Enhanced Interior Gateway Routing Protocol)
	89	
	97	Ethernet-within-IP Encapsulation / RFC3378
	98	Encapsulation Header / RFC1241
	108	IP Payload Compression Protocol / RFC2393
	112	Virtual Router Redundancy Protocol / RFC3768
	134	RSVP-E2E-IGNORE / RFC3175
	135	Mobility Header / RFC3775
	136	UDPLite / RFC3828
	137	MPLS-in-IP / RFC4023
	138	MANET Protocols / RFC-ietf-manet-iana-07.txt
	139-252	Unassigned / IANA
	253	Use for experimentation and testing / RFC3692
	254	Use for experimentation and testing / RFC3692
	255	Reserved / IANA
dhcpsnooping	The source address learned from the DHCP Snooping binding database.	

Parameter(cont.)	**Description**(cont.)	
<source>	The source address of the packets. You can specify a single host, a subnet, or all sources. The following are the valid formats for specifying the source:	
	any	Matches any source IP address.
	host *<ip-addr>*	Matches a single source host with the IP address given by *<ip-addr>* in dotted decimal notation.
	<ip-addr>/*<prefix>*	An IPv4 address, followed by a forward slash, then the prefix length. This matches any source IP address within the specified subnet.
	<ip-addr> *<reverse-mask>*	Alternatively, you can enter a reverse mask in dotted decimal format. For example, entering 192.168.1.1 0.0.0.255 is the same as entering 192.168.1.1/24.
<destination>	The destination address of the packets. You can specify a single host, a subnet, or all destinations. The following are the valid formats for specifying the destination:	
	any	Matches any destination IP address.
	host *<ip-addr>*	Matches a single destination host with the IP address given by *<ip-addr>* in dotted decimal notation.
	<ip-addr>/*<prefix>*	An IPv4 address, followed by a forward slash, then the prefix length. This matches any destination IP address within the specified subnet.
	<ip-addr> *<reverse-mask>*	Alternatively, you can enter a reverse mask in dotted decimal format. For example, entering 192.168.1.1 0.0.0.255 is the same as entering 192.168.1.1/24.
mac	Signifies a MAC and based hardware access-list.	
<mac-source-address>	The source hosts MAC address, entered in HHHH.HHHH.HHHH format.	
<mac-source-mask>	The source hosts MAC wildcard mask entered in HHHH.HHHH.HHHH format. Where Hex FF = Ignore, and Hex 00 = Match.	
any	Matches any source MAC address.	
<mac-destination-address>	The destination hosts MAC address, entered in HHHH.HHHH.HHHH format.	

Parameter(cont.)	**Description**(cont.)
<mac-destination-mask>	The destination hosts wildcard mask entered in HHHH.HHHH.HHHH format. Where Hex `FF` = Ignore, and Hex `00` = Match.
`any`	Matches any destination MAC address.

Mode IPv4 Hardware ACL Configuration

Default Any traffic on an interface controlled by a hardware ACL that does not explicitly match a filter is permitted.

Usage First create a named hardware access-list that applies the appropriate permit, deny requirements etc. Then use the access-group command on page 32.4 to apply this access-list to a specific port or range. Note that this command will apply the access-list only to **incoming** data packets.

An ACL can be configured with multiple ACL filters using sequence numbers. If the sequence number is omitted, the next available multiple of 10 will be used as the sequence number for the new filter. A new ACL filter can be inserted into the middle of an existing list by specifying the appropriate sequence number.

Note The access control list being configured is selected by running the access-list hardware (named) command on page 32.17. with the required access control list number, or name, but with no further parameters selected.

Note Hardware ACLs will **permit** access unless **explicitly denied** by an ACL action.

Examples To add an access-list filter entry to the access-list named `my-list` that will permit any type of IP packet with a source address of `192.168.1.1` and any destination address, use the commands:

```
awplus#                 configure terminal
awplus(config)#         access-list hardware my-list
awplus(config-ip-hw-acl)# permit ip 192.168.1.1/32 any
```

To add an access-list filter entry to the access-list named `my-list` that will permit any type of IP packet with a source address of `192.168.1.1` and a MAC source address of `ffee.ddcc.bbaa` with any IP and MAC destination address, use the commands:

```
awplus#                 configure terminal
awplus(config)#         access-list hardware my-list
awplus(config-ip-hw-acl)# permit ip 192.168.1.1/32 any mac
                          ffee.ddcc.bbaa any
```

To add an access-list filter entry to the access-list named `my-list` a filter that will deny all IGMP packets (protocol 2) from the `192.168.0.0` network with sequence number `50` in access-list, use the commands:

```
             awplus# configure terminal
     awplus(config)# access-list hardware my-list
awplus(config-ip-hw-acl)# 50 deny proto 2 192.168.0.0/16 any
```

Related Commands

access-list hardware (named)
show running-config
show access-list (IPv4 Hardware ACLs)

(access-list hardware MAC filter)

Use this ACL filter to add a MAC filter entry to the current hardware access-list. The filter will match on any IP packet that has the specified source and destination MAC addresses. The parameter **any** may be specified if an address does not matter. If a sequence number is specified, the new filter is inserted at the specified location. Otherwise, the new filter is added at the end of the access-list.

The **no** variant of this command removes a MAC filter entry from the current hardware access-list. You can specify the MAC filter entry for removal by entering either its sequence number (e.g. `no 10`), or by entering its MAC filter profile without specifying its sequence number.

Note that the sequence number can be found by running the show access-list (IPv4 Hardware ACLs) command on page 32.35.

Syntax [mac]

```
[<sequence-number>]
   {deny|permit|send-to-cpu|copy-to-cpu|copy-to-mirror}
   mac {<source-mac-address> <source-mac-mask>|any}
   {<destination-mac-address> <destination-mac-mask>|any}

no {deny|permit|send-to-cpu|copy-to-cpu|copy-to-mirror}
   mac {<source-mac-address> <source-mac-mask>|any}
   {<destination-mac-address> <destination-mac-mask>|any}

no <sequence-number>
```

Parameter	Description
<sequence-number>	`<1-65535>` The sequence number for the filter entry of the selected access control list.
`deny`	Specify packets to reject.
`permit`	Specify packets to accept.
`send-to-cpu`	Specify packets to send to the CPU.
`copy-to-cpu`	Specify packets to copy to the CPU.
`copy-to-mirror`	Specify packets to copy to the CPU.
`mac`	MAC address.
<source-mac-address>	The source MAC address of the packets. Enter this in the format <HHHH.HHHH.HHHH> Where each H is a hexadecimal number that represents a 4 bit binary number.
<source-mac-mask	The mask that will be applied to the source MAC addresses. Enter this in the format <HHHH.HHHH.HHHH> Where each H is a hexadecimal number that represents a 4 bit binary number. For a mask, each value will be either `0` or `F`. Where Hex `FF` = Ignore, and Hex `00` = Match.
`any`	Any source MAC host.
<destination-mac-address>	The destination MAC address of the packets. Enter this in the format <HHHH.HHHH.HHHH> Where each H is a hexadecimal number that represents a 4 bit binary number.

Parameter(cont.)	Description(cont.)
<destination-mac-mask>	The mask that will be applied to the destination MAC addresses. Enter this in the format <HHHH.HHHH.HHHH> Where each H is a hexadecimal number that represents a 4 bit binary number. For a mask, each value will be either 0 or F. Where Hex FF = Ignore, and Hex 00 = Match.
any	Any destination MAC host.

Mode IPv4 Hardware ACL Configuration

Default Any traffic on an interface controlled by a hardware ACL that does not explicitly match a filter is permitted.

Usage First create a named hardware access-list that applies the appropriate permit, deny requirements etc. Then use the access-group command on page 32.4 to apply this access-list to a specific port or range. Note that this command will apply the access-list only to **incoming** data packets.

An ACL can be configured with multiple ACL filters using sequence numbers. If the sequence number is omitted, the next available multiple of 10 will be used as the sequence number for the new filter. A new ACL filter can be inserted into the middle of an existing list by specifying the appropriate sequence number

Note The access control list being configured is selected by running the access-list hardware (named) command on page 32.17. with the required access control list number, or name, but with no further parameters selected.

Note Hardware ACLs will **permit** access unless **explicitly denied** by an ACL action.

Example To add an access-list filter entry to the access-list named `my-list` that will permit packets with a source MAC address of `0000.00ab.1234` and any destination MAC address, use the commands:

```
              awplus#  configure terminal
      awplus(config)#  access-list hardware my-list
awplus(config-ip-hw-acl)#  permit mac 0000.00ab.1234
                           0000.0000.0000 any
```

Example To remove an access-list filter entry that permit packets with a source MAC address of `0000.00ab.1234` and any destination MAC address, use the commands:

```
              awplus#  configure terminal
      awplus(config)#  access-list hardware my-list
awplus(config-ip-hw-acl)#  no permit mac 0000.00ab.1234
                           0000.0000.0000 any
```

Related Commands access-group
access-list hardware (named)
show running-config

(access-list hardware TCP UDP filter)

Use this ACL filter to add a TCP or UDP filter entry to the current hardware access-list. The filter will match on any TCP or UDP type packet that has the specified source and destination IP addresses. The parameter **any** may be specified if an address does not matter. If a sequence number is specified, the new filter is inserted at the specified location. Otherwise, the new filter is added at the end of the access-list.

The **no** variant of this command removes a TCP or UDP filter entry from the current hardware access-list. You can specify the TCP or UDP filter entry for removal by entering either its sequence number (e.g. `no 10`), or by entering its TCP or UDP filter profile without specifying its sequence number.

Note that the sequence number can be found by running the show access-list (IPv4 Hardware ACLs) command on page 32.35.

Syntax [tcp|udp]

```
[<sequence-number>]
    {deny|permit|send-to-cpu|copy-to-cpu|copy-to-mirror}
    {tcp|udp}
    [<source>|
    eq <sourceport>|gt <sourceport>|lt <sourceport>|
    ne <sourceport>|range <start-range> <end-range>]
    [<destination>|
    eq <destport>|gt <destport>|lt <destport>|
    ne <destport>|range <start-range> <end-range>]

no {deny|permit|send-to-cpu|copy-to-cpu|copy-to-mirror}
    {tcp|udp}
    [<source>|
    eq <sourceport>|gt <sourceport>|lt <sourceport>|
    ne <sourceport>|range <start-range> <end-range>]
    [<destination>|
    eq <destport>|gt <destport>|lt <destport>|
    ne <destport>|range <start-range> <end-range>]

no <sequence-number>
```

Parameter	Description
`<sequence-number>`	`<1-65535>` The sequence number for the filter entry of the selected access control list.
`deny`	Access-list rejects packets that match the source and destination filtering specified with this command.
`permit`	Access-list permits packets that match the source and destination filtering specified with this command.
`send-to-cpu`	Specify packets to send to the CPU.
`copy-to-cpu`	Specify packets to copy to the CPU.
`copy-to-mirror`	Specify packets to copy to the mirror port.
`tcp`	TCP packets.
`udp`	UDP packets.

Parameter(cont.)	**Description**(cont.)	
<source>	The source address of the packets. You can specify a single host, a subnet, or all sources. The following are the valid formats for specifying the source:	
	any	Matches any source IP address.
	host <ip-addr>	Matches a single source host with the IP address given by <ip-addr> in dotted decimal notation.
	<ip-addr>/ <prefix>	An IPv4 address, followed by a forward slash, then the prefix length. This matches any source IP address within the specified subnet.
	<ip-addr> <reverse-mask>	Alternatively, you can enter a reverse mask in dotted decimal format. For example, entering 192.168.1.1 0.0.0.255 is the same as entering 192.168.1.1/24.
<sourceport>	The source TCP or UDP port number, specified as an integer between 0 and 65535.	
<destination>	The destination address of the packets. You can specify a single host, a subnet, or all destinations. The following are the valid formats for specifying the destination:	
	any	Matches any destination IP address.
	host <ip-addr>	Matches a single destination host with the IP address given by <ip-addr> in dotted decimal notation.
	<ip-addr>/ <prefix>	An IPv4 address, followed by a forward slash, then the prefix length. This matches any destination IP address within the specified subnet.
	<ip-addr> <reverse-mask>	Alternatively, you can enter a reverse mask in dotted decimal format. For example, entering 192.168.1.1 0.0.0.255 is the same as entering 192.168.1.1/24.
eq	Equal to.	
lt	Less than.	
gt	Greater than.	
ne	Not equal to.	

Parameter(cont.)	**Description**(cont.)
<destport>	The source TCP or UDP port number, specified as an integer between 0 and 65535.
`range`	Specify the range of port numbers between 0 and 65535.
<start-range>	The source or destination port number at the start of the range *<0-65535>*.
<end-range>	The source or destination port number at the end of the range *<0-65535>*.

Mode IPv4 Hardware ACL Configuration

Default Any traffic on an interface controlled by a hardware ACL that does not explicitly match a filter is permitted.

Usage First create a named hardware access-list that applies the appropriate permit, deny requirements etc. Then use the access-group command on page 32.4 to apply this access-list to a specific port or range. Note that this command will apply the access-list only to **incoming** data packets.

An ACL can be configured with multiple ACL filters using sequence numbers. If the sequence number is omitted, the next available multiple of 10 will be used as the sequence number for the new filter. A new ACL filter can be inserted into the middle of an existing list by specifying the appropriate sequence number.

Note The access control list being configured is selected by running the access-list hardware (named) command on page 32.17. with the required access control list number, or name, but with no further parameters selected.

Note Hardware ACLs will **permit** access unless **explicitly denied** by an ACL action.

Example To add an access-list filter entry to access-list named `my-hw-list` that will permit TCP packets with a destination address of `192.168.1.1`, a destination port of `80`, and any source address, and source port, use the commands:

```
               awplus#  configure terminal
       awplus(config)#  access-list hardware my-hw-list
awplus(config-ip-hw-acl)#  permit tcp any 192.168.1.1/32 eq 80
```

Related Commands access-list hardware (named)
show running-config
show access-list (IPv4 Hardware ACLs)

commit (IPv4)

Use this command to commit the ACL filter configuration to hardware immediately without exiting Hardware ACL Configuration mode.

Syntax

```
commit
```

Mode IPv4 Hardware ACL Configuration

Usage Normally, when a hardware ACL is edited, the new configuration state of the ACL is not written to hardware until you exit Hardware ACL Configuration mode. By entering this command you can ensure that the current state of a hardware access-list that is being edited is written to hardware immediately.

Scripts typically do not include the exit command to exit configuration modes, potentially leading to ACL filters in hardware not being correctly updated. Using this **commit** command in a configuration script after specifying a hardware ACL filter ensures that it is updated in the hardware.

Examples To update the hardware with the ACL filter configuration, use the command:

```
              awplus# configure terminal
      awplus(config)# access-list hardware my-hw-list
awplus(config-ip-hw-acl)# commit
```

Related Commands access-list hardware (named)

show access-list (IPv4 Hardware ACLs)

Use this command to display the specified access-list, or all access-lists if none have been specified. Note that only defined access-lists are displayed. An error message is displayed for an undefined access-list.

Syntax

```
show access-list
    [<1-99>|<100-199>|<1300-1999>|<2000-2699>|<3000-3699>|
    <4000-4499>|<access-list-name>]
```

Parameter	Description
<1-99>	IP standard access-list.
<100-199>	IP extended access-list.
<1300-1999>	IP standard access-list (standard - expanded range).
<2000-2699>	IP extended access-list (extended - expanded range).
<3000-3699>	Hardware IP access-list.
<4000-4499>	Hardware MAC access-list.
<access-list-name>	IP named access-list.

Mode User Exec and Privileged Exec

Example To show all access-lists configured on the switch:

```
awplus# show access-list
```

```
Standard IP access list 1
    deny 172.16.2.0, wildcard bits 0.0.0.255
Standard IP access list 20
    deny 192.168.10.0, wildcard bits 0.0.0.255
    deny 192.168.12.0, wildcard bits 0.0.0.255
Hardware IP access list 3001
    permit ip 192.168.20.0 255.255.255.0 any
Hardware IP access list 3020
    permit tcp any 192.0.2.0/24
awplus#show access-list 20
```

Example To show the access-list with an ID of 20:

```
awplus# show access-list 20
```

```
Standard IP access-list 20
    deny 192.168.10.0, wildcard bits 0.0.0.255
    deny 192.168.12.0, wildcard bits 0.0.0.255
```

Note the below error message if you attempt to show an undefined access-list:

```
awplus# show access-list 2
```

```
% Can't find access-list 2
```

Related Commands access-list extended (named)
access-list (hardware MAC numbered)
access-list hardware (named)

show interface access-group

Use this command to display the access groups attached to a port. If an access group is specified, then the output only includes the ports that the specified access group is attached to. If no access group is specified then this command displays all access groups that are attached to the ports that are specified with *<port-list>*.

Note that **access group** is the term given for an access-list when it is applied to an interface.

Note This command will function on your switch in stand-alone mode. but is not supported when the device forms part of a VCStack.

Syntax `show interface <port-list> access-group [<3000-3699>|<4000-4699>`

Parameter	Description
<port-list>	Specify the ports to display information. A port-list can be either: ■ a switch port (e.g. port1.0.12) a static channel group (e.g., sa3) or a dynamic (LACP) channel group (e.g., po3) ■ a continuous range of ports separated by a hyphen, e.g., port1.0.1-1.0.24 or port1.0.1-port1.0.24 or po1-po4 ■ a comma-separated list of ports and port ranges, e.g. port1.0.1,port1.0.3-1.0.24. Do not mix switch ports, static channel groups, and LACP channel groups in the same list.
`access group`	Select the access group whose details you want to show.
<3000-3699>	Specifies the Hardware IP access-list.
<4000-4699>	Specifies the Hardware MAC access-list.

Mode User Exec and Privileged Exec

Example To show all access-lists attached to `port1.0.1`, use the command:

```
awplus# show interface port1.0.1 access-group
```

Output Figure 32-1: Example output from the **show interface access-group** command

```
Interface port1.0.1
  access-group 3000
  access-group 3002
  access-group 3001
```

Related Commands access-group

Chapter 33: IPv4 Software Access Control List (ACL) Commands

Introduction

This chapter provides an alphabetical reference for the IPv4 Software Access Control List (ACL) commands, and contains detailed command information and command examples about IPv4 software ACLs as applied to Routing and Multicasting, which are not applied to interfaces.

Note See Chapter 31, Access Control Lists Introduction for descriptions of ACLs, and for further information about rules when applying ACLs see the ACL Rules section.

See ACL Filter Sequence Numbers and ACL Filter Sequence Number Behavior sections in Chapter 31, Access Control Lists Introduction about ACL Filters.

See all relevant Routing commands and configurations in "IPv4 Software Access Control List (ACL) Commands" and all relevant Multicast commands and configurations in "Multicast Applications".

To apply ACLs to an LACP channel group, apply it to all the individual switch ports in the channel group. To apply ACLs to a static channel group, apply it to the static channel group itself. For more information on link aggregation see Chapter 20, Link Aggregation Introduction and Configuration, and Chapter 21, Link Aggregation Commands.

Note Text in parenthesis in command names indicates usage not keyword entry. For example, **access-list hardware (named)** indicates named IPv4 hardware ACLs entered as `access-list hardware <name>` where `<name>` is a placeholder not a keyword.

Note Parenthesis surrounding ACL filters indicates the type of ACL filter not the keyword entry in the CLI, such as **(access-list standard numbered filter)** represents command entry in the format shown in the syntax `[<sequence-number>] {deny|permit} {<source>|host <host-address>|any}`.

Note Software ACLs will **deny** access unless **explicitly permitted** by an ACL action.

IPv4 Software Access List Commands and Prompts

Many of the ACL commands operate from sub-modes that are specific to particular ACL types. The table "IPv4 Software Access List Commands and Prompts" shows the CLI prompts at which ACL commands are entered.

Table 33-1: IPv4 Software Access List Commands and Prompts

Command Name	Command Mode	Prompt
clear ip prefix-list	Privileged Exec	`awplus#`
show ip access-list	Privileged Exec	`awplus#`
access-group	Global Configuration	`awplus(config)#`
access-list (extended numbered)	Global Configuration	`awplus(config)#`
access-list standard (named)	Global Configuration	`awplus(config)#`
access-list (standard numbered)	Global Configuration	`awplus(config)#`
ip prefix-list	Global Configuration	`awplus(config)#`
maximum-access-list	Global Configuration	`awplus(config)#`
dos	Interface Configuration	`awplus(config-if)#`
(access-list extended ICMP filter)	IPv4 Extended ACL Configuration	`awplus(config-ip-ext-acl)#`
(access-list extended IP filter)	IPv4 Extended ACL Configuration	`awplus(config-ip-ext-acl)#`
(access-list extended IP protocol filter)	IPv4 Extended ACL Configuration	`awplus(config-ip-ext-acl)#`
(access-list extended TCP UDP filter)	IPv4 Extended ACL Configuration	`awplus(config-ip-ext-acl)#`
(access-list standard named filter)	IPv4 Standard ACL Configuration	`awplus(config-ip-std-acl)#`
(access-list standard numbered filter)	IPv4 Standard ACL Configuration	`awplus(config-ip-std-acl)#`

Command List

access-list extended (named)

This command configures an extended named access-list that permits or denies packets from specific source and destination IP addresses. You can either create an extended named ACL together with an ACL filter entry in the Global Configuration mode, or you can use the IPv4 Extended ACL Configuration mode for sequenced ACL filter entry after entering a list name.

The **no** variant of this command removes a specified extended named access-list.

Syntax [list-name]

```
access-list extended <list-name>

no access-list extended <list-name>
```

Parameter	Description
<list-name>	A user-defined name for the access-list

Syntax [icmp]

```
access-list extended <list-name>
    {deny|permit}
    icmp <source> <destination>
    [icmp-type <type-number>]
    [log]

no access-list extended <list-name>
    {deny|permit}
    icmp <source> <destination>
    [icmp-type <type-number>]
    [log]
```

Table 33-2: Parameters in the access-list extended (named) command - icmp

Parameter	Description
<list-name>	A user-defined name for the access-list.
deny	The access-list rejects packets that match the type, source, and destination filtering specified with this command.
permit	The access-list permits packets that match the type, source, and destination filtering specified with this command.
icmp	The access-list matches only ICMP packets.
icmp-type	Matches only a specified type of ICMP messages. This is valid only when the filtering is set to match ICMP packets.

Table 33-2: Parameters in the access-list extended (named) command - icmp(cont.)

Parameter(cont.)	**Description**(cont.)	
<source>	The source address of the packets. You can specify a single host, a subnet, or all sources. The following are the valid formats for specifying the source:	
	any	Matches any source IP address.
	host *<ip-addr>*	Matches a single source host with the IP address given by *<ip-addr>* in dotted decimal notation.
	<ip-addr>/ *<prefix>*	An IPv4 address, followed by a forward slash, then the prefix length. This matches any source IP address within the specified subnet.
	<ip-addr> *<reverse-mask>*	Alternatively, you can enter a reverse mask in dotted decimal format. For example, entering 192.168.1.1 0.0.0.255 is the same as entering 192.168.1.1/24.
<destination>	The destination address of the packets. You can specify a single host, a subnet, or all destinations. The following are the valid formats for specifying the destination:	
	any	Matches any destination IP address.
	host *<ip-addr>*	Matches a single destination host with the IP address given by *<ip-addr>* in dotted decimal notation.
	<ip-addr>/ *<prefix>*	An IPv4 address, followed by a forward slash, then the prefix length. This matches any destination IP address within the specified subnet.
	<ip-addr> *<reverse-mask>*	Alternatively, you can enter a reverse mask in dotted decimal format. For example, entering 192.168.1.1 0.0.0.255 is the same as entering 192.168.1.1/24.

Table 33-2: Parameters in the access-list extended (named) command - icmp(cont.)

Parameter(cont.)	**Description**(cont.)	
`<type-number>`	The ICMP type, as defined in RFC792 and RFC950. Specify one of the following integers to create a filter for the ICMP message type:	
	0	Echo replies.
	3	Destination unreachable messages.
	4	Source quench messages.
	5	Redirect (change route) messages.
	8	Echo requests.
	11	Time exceeded messages.
	12	Parameter problem messages.
	13	Timestamp requests.
	14	Timestamp replies.
	15	Information requests.
	16	Information replies.
	17	Address mask requests.
	18	Address mask replies.
`log`	Logs the results.	

Syntax [tcp|udp]

```
access-list extended <list-name>
    {deny|permit}
    {tcp|udp}
    <source>
    [eq <sourceport>|lt <sourceport>|gt <sourceport>|ne <sourceport>]
    <destination>
    [eq <destport>|lt <destport>|gt <destport>|ne <destport>]
    [log]]

no access-list extended <list-name>
    {deny|permit}
    {tcp|udp}
    <source>
    [eq <sourceport>|lt <sourceport>|gt <sourceport>|ne <sourceport>]
    <destination>
    [eq <destport>|lt <destport>|gt <destport>|ne <destport>]
    [log]]
```

Table 33-3: Parameters in the access-list extended (named) command - tcp|udp

Parameter	Description	
`<list-name>`	A user-defined name for the access-list.	
`deny`	The access-list rejects packets that match the type, source, and destination filtering specified with this command.	
`permit`	The access-list permits packets that match the type, source, and destination filtering specified with this command.	
`tcp`	The access-list matches only TCP packets.	
`udp`	The access-list matches only UDP packets.	
`<source>`	The source address of the packets. You can specify a single host, a subnet, or all sources. The following are the valid formats for specifying the source:	
	`any`	Matches any source IP address.
	`host <ip-addr>`	Matches a single source host with the IP address given by `<ip-addr>` in dotted decimal notation.
	`<ip-addr>/<prefix>`	An IPv4 address, followed by a forward slash, then the prefix length. This matches any source IP address within the specified subnet.
	`<ip-addr> <reverse-mask>`	Alternatively, you can enter a reverse mask in dotted decimal format. For example, entering `192.168.1.1 0.0.0.255` is the same as entering `192.168.1.1/24`.

Table 33-3: Parameters in the access-list extended (named) command - tcp|udp(cont.)

Parameter(cont.)	**Description**(cont.)	
<destination>	The destination address of the packets. You can specify a single host, a subnet, or all destinations. The following are the valid formats for specifying the destination:	
	any	Matches any destination IP address.
	host *<ip-addr>*	Matches a single destination host with the IP address given by *<ip-addr>* in dotted decimal notation.
	<ip-addr>/ *<prefix>*	An IPv4 address, followed by a forward slash, then the prefix length. This matches any destination IP address within the specified subnet.
	<ip-addr> *<reverse-mask>*	Alternatively, you can enter a reverse mask in dotted decimal format. For example, entering 192.168.1.1 0.0.0.255 is the same as entering 192.168.1.1/24.
<sourceport>	The source port number, specified as an integer between 0 and 65535.	
<destport>	The destination port number, specified as an integer between 0 and 65535.	
eq	Matches port numbers equal to the port number specified immediately after this parameter.	
lt	Matches port numbers less than the port number specified immediately after this parameter.	
gt	Matches port numbers greater than the port number specified immediately after this parameter.	
ne	Matches port numbers not equal to the port number specified immediately after this parameter.	
log	Log the results.	

Syntax [proto|any|ip]

```
access-list extended <list-name>
    {deny|permit}
    {proto <ip-protocol>|any|ip}
    {<source>}
    {<destination>}
    [log]
```

```
no access-list extended <list-name>
    {deny|permit}
    {proto <ip-protocol>|any|ip}
    {<source>}
    {<destination>}
    [log]
```

Table 33-4: Parameters in the access-list extended (named) command - proto|ip|any

<table>
<tr><th>Parameter</th><th colspan="2">Description</th></tr>
<tr><td><code>&lt;list-name&gt;</code></td><td colspan="2">A user-defined name for the access-list.</td></tr>
<tr><td><code>deny</code></td><td colspan="2">The access-list rejects packets that match the type, source, and destination filtering specified with this command.</td></tr>
<tr><td><code>permit</code></td><td colspan="2">The access-list permits packets that match the type, source, and destination filtering specified with this command.</td></tr>
<tr><td><code>proto</code></td><td colspan="2">Matches only a specified type of IP Protocol.</td></tr>
<tr><td><code>any</code></td><td colspan="2">The access-list matches any type of IP packet.</td></tr>
<tr><td><code>ip</code></td><td colspan="2">The access-list matches only IP packets.</td></tr>
<tr><td rowspan="5"><code>&lt;source&gt;</code></td><td colspan="2">The source address of the packets. You can specify a single host, a subnet, or all sources. The following are the valid formats for specifying the source:</td></tr>
<tr><td><code>any</code></td><td>Matches any source IP address.</td></tr>
<tr><td><code>host &lt;ip-addr&gt;</code></td><td>Matches a single source host with the IP address given by <code>&lt;ip-addr&gt;</code> in dotted decimal notation.</td></tr>
<tr><td><code>&lt;ip-addr&gt;/</code>
<code>&lt;prefix&gt;</code></td><td>An IPv4 address, followed by a forward slash, then the prefix length. This matches any source IP address within the specified subnet.</td></tr>
<tr><td><code>&lt;ip-addr&gt;</code>
<code>&lt;reverse-mask&gt;</code></td><td>Alternatively, you can enter a reverse mask in dotted decimal format. For example, entering <code>192.168.1.1 0.0.0.255</code> is the same as entering <code>192.168.1.1/24</code>.</td></tr>
</table>

Table 33-4: Parameters in the access-list extended (named) command - proto|ip|any(cont.)

Parameter(cont.)	Description(cont.)	
<destination>	The destination address of the packets. You can specify a single host, a subnet, or all destinations. The following are the valid formats for specifying the destination:	
	any	Matches any destination IP address.
	host *<ip-addr>*	Matches a single destination host with the IP address given by *<ip-addr>* in dotted decimal notation.
	<ip-addr>/ *<prefix>*	An IPv4 address, followed by a forward slash, then the prefix length. This matches any destination IP address within the specified subnet.
	<ip-addr> *<reverse-mask>*	Alternatively, you can enter a reverse mask in dotted decimal format. For example, entering 192.168.1.1 0.0.0.255 is the same as entering 192.168.1.1/24.
log	Logs the results.	
<ip-protocol>	The IP protocol number, as defined by IANA (Internet Assigned Numbers Authority http://www.iana.org/assignments/protocol-numbers)	
	Protocol Number	**Protocol Description [RFC Reference]**
	1	Internet Control Message [RFC792]
	2	Internet Group Management [RFC1112]
	3	Gateway-to-Gateway [RFC823]
	4	IP in IP [RFC2003]
	5	Stream [RFC1190] [RFC1819]
	6	TCP (Transmission Control Protocol) [RFC793]
	8	EGP (Exterior Gateway Protocol) [RFC888]
	9	IGP (Interior Gateway Protocol) [IANA]
	11	Network Voice Protocol [RFC741]
	17	UDP (User Datagram Protocol) [RFC768]
	20	Host monitoring [RFC869]
	27	RDP (Reliable Data Protocol) [RFC908]
	28	IRTP (Internet Reliable Transaction Protocol) [RFC938]
	29	ISO-TP4 (ISO Transport Protocol Class 4) [RFC905]
	30	Bulk Data Transfer Protocol [RFC969]

Table 33-4: Parameters in the access-list extended (named) command - proto|ip|any(cont.)

Parameter(cont.)	Description(cont.)	
<ip-protocol> (cont.)	**Protocol Number**	**Protocol Description [RFC Reference]**
	33	Datagram Congestion Control Protocol [RFC4340]
	48	DSR (Dynamic Source Routing Protocol) [RFC4728]
	50	ESP (Encap Security Payload) [RFC2406]
	51	AH (Authentication Header) [RFC2402]
	54	NARP (NBMA Address Resolution Protocol) [RFC1735]
	88	EIGRP (Enhanced Interior Gateway Routing Protocol)
	89	OSPFIGP [RFC1583]
	97	Ethernet-within-IP Encapsulation / RFC3378
	98	Encapsulation Header / RFC1241
	108	IP Payload Compression Protocol / RFC2393
	112	Virtual Router Redundancy Protocol / RFC3768
	134	RSVP-E2E-IGNORE / RFC3175
	135	Mobility Header / RFC3775
	136	UDPLite / RFC3828
	137	MPLS-in-IP / RFC4023
	138	MANET Protocols / RFC-ietf-manet-iana-07.txt
	139-252	Unassigned / IANA
	253	Use for experimentation and testing / RFC3692
	254	Use for experimentation and testing / RFC3692
	255	Reserved / IANA

Mode Global Configuration

Default Any traffic controlled by a software ACL that does not explicitly match a filter is denied.

Usage Use this command when configuring access-list for filtering IP software packets. To enable backwards compatibility you can either create access-lists from within this command, or you can enter **access-list** followed by only the number. This latter method moves you to the IPv4 Extended ACL Configuration mode for the selected access-list number, and from here you can configure your access-lists by using the commands (access-list extended ICMP filter), (access-list extended IP filter), and (access-list extended IP protocol filter).

The table "IPv4 Software Access List Commands and Prompts" on page 33.3 shows the prompts at which ACL commands are entered. See the relevant links shown for the **Related Commands**.

Note that packets must match both the source and the destination details.

Note Software ACLs will **deny** access unless **explicitly permitted** by an ACL action.

Example You can enter the extended named ACL in the Global Configuration mode together with the ACL filter entry on the same line, as in previous software releases as shown below:

```
        awplus#  configure terminal

awplus(config)#  access-list extended TK deny tcp 2.2.2.3/24 eq
                 14 3.3.3.4/24 lt 12 log
```

Alternatively, you can enter the extended named ACL in Global Configuration mode before specifying the ACL filter entry in the IPv4 Extended ACL Configuration mode, as shown below:

```
                awplus#  configure terminal

        awplus(config)#  access-list extended TK

awplus(config-ip-ext-acl)#  deny tcp 2.2.2.3/24 eq 14 3.3.3.4/24
                         lt 12 log
```

Related Commands (access-list extended ICMP filter)
(access-list extended IP filter)
(access-list extended TCP UDP filter)
show running-config
show ip access-list

access-list (extended numbered)

This command configures an extended numbered access-list that permits or denies packets from specific source and destination IP addresses. You can either create an extended numbered ACL together with an ACL filter entry in the Global Configuration mode, or you can use the IPv4 Extended ACL Configuration mode for sequenced ACL filter entry after entering a list number.

The **no** variant of this command removes a specified extended named access-list.

Syntax [list-number]

```
access-list {<100-199>|<2000-2699>}

no access-list {<100-199>|<2000-2699>}
```

Parameter	Description
<100-199>	IP extended access-list.
<2000-2699>	IP extended access-list (expanded range).

Syntax [deny|permit]

```
access-list {<100-199>|<2000-2699>}
    {deny|permit}
    ip <source> <destination>

no access-list {<100-199>|<2000-2699>}
    {deny|permit}
    ip <source> <destination>
```

Parameter	Description	
<100-199>	IP extended access-list.	
<2000-2699>	IP extended access-list (expanded range).	
`deny`	Access-list rejects packets that match the source and destination filtering specified with this command.	
`permit`	Access-list permits packets that match the source and destination filtering specified with this command.	
<source>	The source address of the packets. You can specify a single host, a subnet, or all sources. The following are the valid formats for specifying the source:	
	`any`	Matches any source IP address.
	`host` *<ip-addr>*	Matches a single source host with the IP address given by *<ip-addr>* in dotted decimal notation.
	<ip-addr> *<reverse-mask>*	An IPv4 address, followed by a reverse mask in dotted decimal format. For example, entering `192.168.1.1 0.0.0.255` is the same as entering `192.168.1.1/24`. This matches any source IP address within the specified subnet.
<destination>	The destination address of the packets. You can specify a single host, a subnet, or all destinations. The following are the valid formats for specifying the destination:	

Parameter(cont.)	Description(cont.)	
	any	Matches any destination IP address.
	host <*ip-addr*>	Matches a single destination host with the IP address given by <*ip-addr*> in dotted decimal notation.
	<*ip-addr*> <*reverse-mask*>	An IPv4 address, followed by a reverse mask in dotted decimal format. For example, entering 192.168.1.1 0.0.0.255 is the same as entering 192.168.1.1/24. This matches any destination IP address within the specified subnet.

Mode Global Configuration

Default Any traffic controlled by a software ACL that does not explicitly match a filter is denied.

Usage Use this command when configuring access-list for filtering IP software packets. To enable backwards compatibility you can either create access-lists from within this command, or you can enter **access-list** followed by only the number. This latter method moves you to the IPv4 Extended ACL Configuration mode for the selected access-list number, and from here you can configure your access-lists by using the commands (access-list extended ICMP filter), (access-list extended IP filter), and (access-list extended IP protocol filter).

The table "IPv4 Software Access List Commands and Prompts" on page 33.3 shows the prompts at which ACL commands are entered. See the relevant links shown for the **Related Commands**.

Note that packets must match both the source and the destination details.

Note Software ACLs will **deny** access unless **explicitly permitted** by an ACL action.

Example You can enter the extended named ACL in the Global Configuration mode together with the ACL filter entry on the same line, as in previous software releases as shown below:

```
        awplus#  configure terminal
awplus(config)#  access-list 101 deny ip 172.16.10.0 0.0.0.255
                 any
```

Alternatively, you can enter the extended named ACL in Global Configuration mode before specifying the ACL filter entry in the IPv4 Extended ACL Configuration mode, as shown below:

```
               awplus#  configure terminal
       awplus(config)#  access-list 101
awplus(config-ip-ext-acl)#  deny ip 172.16.10.0 0.0.0.255 any
```

Related Commands

(access-list extended ICMP filter)
(access-list extended IP filter)
(access-list extended TCP UDP filter)
show running-config
show ip access-list

(access-list extended ICMP filter)

Use this ACL filter to add a new ICMP filter entry to the current extended access-list. If the sequence number is specified, the new filter is inserted at the specified location. Otherwise, the new filter is added at the end of the access-list.

The **no** variant of this command removes an ICMP filter entry from the current extended access-list. You can specify the ICMP filter entry for removal by entering either its sequence number (e.g. `no 10`), or by entering its ICMP filter profile without specifying its sequence number.

Note that the sequence number can be found by running the show access-list (IPv4 Software ACLs) command.

Syntax [icmp]

```
[<sequence-number>] {deny|permit}
    icmp <source> <destination>
    [icmp-type <icmp-value>] [log]

no {deny|permit} icmp <source> <destination>
    [icmp-type <icmp-value>] [log]

no <sequence-number>
```

<table>
<tr><th>Parameter</th><th colspan="2">Description</th></tr>
<tr><td><i>&lt;sequence-number&gt;</i></td><td colspan="2">&lt;1-65535>
The sequence number for the filter entry of the selected access control list.</td></tr>
<tr><td>deny</td><td colspan="2">Access-list rejects packets that match the source and destination filtering specified with this command.</td></tr>
<tr><td>permit</td><td colspan="2">Access-list permits packets that match the source and destination filtering specified with this command.</td></tr>
<tr><td>icmp</td><td colspan="2">ICMP packet type.</td></tr>
<tr><td rowspan="3"><i>&lt;source&gt;</i></td><td colspan="2">The source address of the packets. You can specify a single host, a subnet, or all sources. The following are the valid formats for specifying the source:</td></tr>
<tr><td><i>&lt;ip-addr>/<prefix&gt;</i></td><td>An IPv4 address, followed by a forward slash, then the prefix length. This matches any source IP address within the specified subnet.</td></tr>
<tr><td>any</td><td>Matches any source IP address.</td></tr>
<tr><td rowspan="3"><i>&lt;destination&gt;</i></td><td colspan="2">The destination address of the packets. You can specify a single host, a subnet, or all destinations. The following are the valid formats for specifying the destination:</td></tr>
<tr><td><i>&lt;ip-addr>/<prefix&gt;</i></td><td>An IPv4 address, followed by a forward slash, then the prefix length. This matches any destination IP address within the specified subnet.</td></tr>
<tr><td>any</td><td>Matches any destination IP address.</td></tr>
<tr><td>icmp-type</td><td colspan="2">The ICMP type.</td></tr>
</table>

Parameter(cont.)	**Description**(cont.)
<*icmp-value*>	The value of the ICMP type.
log	Log the results.

Mode IPv4 Extended ACL Configuration

Default Any traffic controlled by a software ACL that does not explicitly match a filter is denied.

Usage An ACL can be configured with multiple ACL filters using sequence numbers. If the sequence number is omitted, the next available multiple of 10 will be used as the sequence number for the new filter. A new ACL filter can be inserted into the middle of an existing list by specifying the appropriate sequence number.

Note The access control list being configured is selected by running the access-list (extended numbered) command or the access-list extended (named) command, with the required access control list number, or name - but with no further parameters selected.

Note Software ACLs will **deny** access unless **explicitly permitted** by an ACL action.

Examples To add a new entry in access-list called `my-list` that will reject ICMP packets from `10.0.0.1` to `192.168.1.1`, use the commands:

```
                 awplus# configure terminal
         awplus(config)# access-list extended my-list
awplus(config-ip-ext-acl)# deny icmp 10.0.0.1/32 192.168.1.1/32
```

Use the following commands to add a new filter at sequence number 5 position of the access-list called `my-list`. The filter will accept the ICMP type 8 packets from `10.1.1.0/24` network, to `192.168.1.0` network:

```
                 awplus# configure terminal
         awplus(config)# access-list extended my-list
awplus(config-ip-ext-acl)# 5 permit icmp 10.1.1.0/24
                           192.168.1.0/24 icmp-type 8
```

Related Commands access-group
show running-config
show ip access-list

(access-list extended IP filter)

Use this ACL filter to add a new IP filter entry to the current extended access-list. If the sequence number is specified, the new filter is inserted at the specified location. Otherwise, the new filter is added at the end of the access-list.

The **no** variant of this command removes an IP filter entry from the current extended access-list. You can specify the IP filter entry for removal by entering either its sequence number (e.g. `no 10`), or by entering its IP filter profile without specifying its sequence number.

Note that the sequence number can be found by running the show access-list (IPv4 Software ACLs) command.

Syntax [ip]

```
[<sequence-number>] {deny|permit} ip <source> <destination>

no {deny|permit} ip <source> <destination>

no <sequence-number>
```

<table>
<tr><th>Parameter</th><th colspan="2">Description</th></tr>
<tr><td><i>&lt;sequence-number&gt;</i></td><td colspan="2">&lt;1-65535>
The sequence number for the filter entry of the selected access control list.</td></tr>
<tr><td>deny</td><td colspan="2">Access-list rejects packets that match the source and destination filtering specified with this command.</td></tr>
<tr><td>permit</td><td colspan="2">Access-list permits packets that match the source and destination filtering specified with this command.</td></tr>
<tr><td><i>&lt;source&gt;</i></td><td colspan="2">The source address of the packets. You can specify a single host, a subnet, or all sources. The following are the valid formats for specifying the source:</td></tr>
<tr><td></td><td>any</td><td>Matches any source IP address.</td></tr>
<tr><td></td><td>host <i>&lt;ip-addr&gt;</i></td><td>Matches a single source host with the IP address given by <i>&lt;ip-addr&gt;</i> in dotted decimal notation.</td></tr>
<tr><td></td><td><i>&lt;ip-addr></i>
<i>&lt;reverse-mask&gt;</i></td><td>Alternatively, enter an IPv4 address followed by a reverse mask in dotted decimal format. For example, enter 192.168.1.1 0.0.0.255.</td></tr>
<tr><td><i>&lt;destination&gt;</i></td><td colspan="2">The destination address of the packets. You can specify a single host, a subnet, or all destinations. The following are the valid formats for specifying the destination:</td></tr>
<tr><td></td><td>any</td><td>Matches any destination IP address.</td></tr>
<tr><td></td><td>host <i>&lt;ip-addr&gt;</i></td><td>Matches a single destination host with the IP address given by <i>&lt;ip-addr&gt;</i> in dotted decimal notation.</td></tr>
<tr><td></td><td><i>&lt;ip-addr></i>
<i>&lt;reverse-mask&gt;</i></td><td>Alternatively, enter an IPv4 address followed by a reverse mask in dotted decimal format. For example, enter 192.168.1.1 0.0.0.255.</td></tr>
</table>

Mode Extended ACL Configuration

Default Any traffic controlled by a software ACL that does not explicitly match a filter is denied.

Usage An ACL can be configured with multiple ACL filters using sequence numbers. If the sequence number is omitted, the next available multiple of 10 will be used as the sequence number for the new filter. A new ACL filter can be inserted into the middle of an existing list by specifying the appropriate sequence number.

Note The access control list being configured is selected by running the access-list (extended numbered) command or the access-list extended (named) command, with the required access control list number, or name - but with no further parameters selected.

Note Software ACLs will **deny** access unless **explicitly permitted** by an ACL action.

Example 1 [list-number] First use the following commands to enter the IPv4 Extended ACL Configuration mode and define a numbered extended access-list 101:

```
                 awplus# configure terminal
         awplus(config)# access-list 101
awplus(config-ip-ext-acl)#
```

Then use the following commands to add a new entry to the numbered extended access-list 101 that will reject packets from 10.0.0.1 to 192.168.1.1:

```
awplus(config-ip-ext-acl)# deny ip host 10.0.0.1 host
                           192.168.1.1
awplus(config-ip-ext-acl)# 20 permit ip any any
```

Example 2 [list-name] First use the following commands to enter the IPv4 Extended ACL Configuration mode and define a named access-list called my-acl:

```
                 awplus# configure terminal
         awplus(config)# access-list extended my-acl
awplus(config-ip-ext-acl)#
```

Then use the following commands to add a new entry to the named access-list my-acl that will reject packets from 10.0.0.1 to 192.168.1.1:

```
awplus(config-ip-ext-acl)# deny ip host 10.0.0.1 host
                           192.168.1.1
awplus(config-ip-ext-acl)# 20 permit ip any any
```

Example 3 [list-number]

Use the following commands to remove the access-list filter entry with sequence number 20 from extended numbered access-list 101.

```
              awplus# configure terminal
      awplus(config)# access-list 101
awplus(config-ip-ext-acl)# no 20
```

Example 4 [list-name]

Use the following commands to remove the access-list filter entry with sequence number 20 from extended named access-list my-acl:.

```
              awplus# configure terminal
      awplus(config)# access-list extended my-acl
awplus(config-ip-ext-acl)# no 20
```

Related Commands

access-list extended (named)
access-list (extended numbered)
show running-config
show ip access-list

(access-list extended IP protocol filter)

Use this ACL filter to add a new IP protocol type filter entry to the current extended access-list. If the sequence number is specified, the new filter is inserted at the specified location. Otherwise, the new filter is added at the end of the access-list.

The **no** variant of this command removes an IP protocol filter entry from the current extended access-list. You can specify the IP filter entry for removal by entering either its sequence number (e.g. `no 10`), or by entering its IP filter profile without specifying its sequence number.

Note that the sequence number can be found by running the show access-list (IPv4 Software ACLs) command.

Syntax [proto]

```
[<sequence-number>] {deny|permit} proto <ip-protocol>
   <source> <destination> [log]
```

```
no {deny|permit} proto <ip-protocol> <source> <destination> [log]
```

```
no <sequence-number>
```

Parameter	Description
<sequence-number>	<1-65535> The sequence number for the filter entry of the selected access control list.
`deny`	Access-list rejects packets that match the source and destination filtering specified with this command.
`permit`	Access-list permits packets that match the source and destination filtering specified with this command.
`proto` *<ip-protocol>*	The IP Protocol type specified by its protocol number <1-255>.
<ip-protocol>	The IP protocol number, as defined by IANA (Internet Assigned Numbers Authority http://www.iana.org/assignments/protocol-numbers).

Protocol Number	Protocol Description [RFC Reference]
1	Internet Control Message [RFC792]
2	Internet Group Management [RFC1112]
3	Gateway-to-Gateway [RFC823]
4	IP in IP [RFC2003]
5	Stream [RFC1190] [RFC1819]
6	TCP (Transmission Control Protocol) [RFC793]
8	EGP (Exterior Gateway Protocol) [RFC888]
9	IGP (Interior Gateway Protocol) [IANA]
11	Network Voice Protocol [RFC741]

Parameter(cont.)	**Description**(cont.)	

`<ip-protocol>`
`(cont.)`
	17	UDP (User Datagram Protocol) [RFC768]

	20	Host monitoring [RFC869]
	27	RDP (Reliable Data Protocol) [RFC908]
	28	IRTP (Internet Reliable Transaction Protocol) [RFC938]
	29	ISO-TP4 (ISO Transport Protocol Class 4) [RFC905]
	30	Bulk Data Transfer Protocol [RFC969]
	33	DCCP (Datagram Congestion Control Protocol) [RFC4340]
	48	DSR (Dynamic Source Routing Protocol) [RFC4728]
	50	ESP (Encap Security Payload) [RFC2406]
	51	AH (Authentication Header) [RFC2402]
	54	NARP (NBMA Address Resolution Protocol) [RFC1735]
	88	EIGRP (Enhanced Interior Gateway Routing Protocol)
	89	OSPFIGP [RFC1583]
	97	Ethernet-within-IP Encapsulation / RFC3378
	98	Encapsulation Header / RFC1241
	108	IP Payload Compression Protocol / RFC2393
	112	Virtual Router Redundancy Protocol / RFC3768
	134	RSVP-E2E-IGNORE / RFC3175
	135	Mobility Header / RFC3775
	136	UDPLite / RFC3828
	137	MPLS-in-IP / RFC4023
	138	MANET Protocols / RFC-ietf-manet-iana-07.txt
	139-252	Unassigned / IANA
	253	Use for experimentation and testing / RFC3692
	254	Use for experimentation and testing / RFC3692
	255	Reserved / IANA

Parameter(cont.)	**Description**(cont.)	
<source>	The source address of the packets. You can specify a single host, a subnet, or all sources. The following are the valid formats for specifying the source:	
	<ip-addr>/<prefix>	An IPv4 address, followed by a forward slash, then the prefix length. This matches any source IP address within the specified subnet.
	`any`	Matches any source IP address.
<destination>	The destination address of the packets. You can specify a single host, a subnet, or all destinations. The following are the valid formats for specifying the destination:	
	<ip-addr>/<prefix>	An IPv4 address, followed by a forward slash, then the prefix length. This matches any destination IP address within the specified subnet.
	`any`	Matches any destination IP address.
`log`	Log the results.	

Mode IPv4 Extended ACL Configuration

Default Any traffic controlled by a software ACL that does not explicitly match a filter is denied.

Usage An ACL can be configured with multiple ACL filters using sequence numbers. If the sequence number is omitted, the next available multiple of 10 will be used as the sequence number for the new filter. A new ACL filter can be inserted into the middle of an existing list by specifying the appropriate sequence number.

Note The access control list being configured is selected by running the access-list (extended numbered) command or the access-list extended (named) command, with the required access control list number, or name - but with no further parameters selected.

Note Software ACLs will **deny** access unless **explicitly permitted** by an ACL action.

Example 1 [creating a list] Use the following commands to add a new access-list filter entry to the access-list named `my-list` that will reject IP packets from source address `10.10.1.1/32` to destination address `192.68.1.1/32`:

```
                awplus# configure terminal
        awplus(config)# access-list extended my-list
awplus(config-ip-ext-acl)# deny ip 10.10.1.1/32 192.168.1.1/32
```

Example 2 [adding to a list]

Use the following commands to add a new access-list filter entry at sequence position 5 in the access-list named `my-list` that will accept packets from source address `10.10.1.1/24` to destination address `192.68.1.1/24`:

```
                  awplus# configure terminal
          awplus(config)# access-list extended my-list
awplus(config-ip-ext-acl)# 5 permit ip 10.10.1.1/24 192.168.1.1/
                           24
```

Related Commands

access-list extended (named)
access-list (extended numbered)
show running-config
show ip access-list

(access-list extended TCP UDP filter)

Use this ACL filter to add a new TCP or UDP filter entry to the current extended access-list. If the sequence number is specified, the new filter is inserted at the specified location. Otherwise, the new filter is added at the end of the access-list.

The **no** variant of this command removes a TCP or UDP filter entry from the current extended access-list. You can specify the TCP or UDP filter entry for removal by entering either its sequence number (e.g. `no 10`), or by entering its TCP or UDP filter profile without specifying its sequence number.

Note that the sequence number can be found by running the show access-list (IPv4 Software ACLs) command.

Syntax [tcp|udp]

```
[<sequence-number>] {deny|permit} {tcp|udp}
   <source>
   {eq <sourceport>|lt <sourceport>|gt <sourceport>|ne <sourceport>}
   <destination>
   [eq <destport>|lt <destport>|gt <destport>|ne <destport>]
   [log]

no {deny|permit} {tcp|udp}
   <source>
   {eq <sourceport>|lt <sourceport>|gt <sourceport>|ne <sourceport>}
   <destination>
   [eq <destport>|lt <destport>|gt <destport>|ne <destport>]
   [log]

no <sequence-number>
```

<table>
<tr><th>Parameter</th><th colspan="2">Description</th></tr>
<tr><td><i>&lt;sequence-number&gt;</i></td><td colspan="2">&lt;1-65535>
The sequence number for the filter entry of the selected access control list.</td></tr>
<tr><td>deny</td><td colspan="2">Access-list rejects packets that match the source and destination filtering specified with this command.</td></tr>
<tr><td>permit</td><td colspan="2">Access-list permits packets that match the source and destination filtering specified with this command.</td></tr>
<tr><td>tcp</td><td colspan="2">The access-list matches only TCP packets.</td></tr>
<tr><td>udp</td><td colspan="2">The access-list matches only UDP packets.</td></tr>
<tr><td rowspan="3"><i>&lt;source&gt;</i></td><td colspan="2">The source address of the packets. You can specify a single host, a subnet, or all sources. The following are the valid formats for specifying the source:</td></tr>
<tr><td><i>&lt;ip-addr>/<prefix&gt;</i></td><td>An IPv4 address, followed by a forward slash, then the prefix length. This matches any source IP address within the specified subnet.</td></tr>
<tr><td>any</td><td>Matches any source IP address.</td></tr>
<tr><td><i>&lt;sourceport&gt;</i></td><td colspan="2">The source port number, specified as an integer between 0 and 65535.</td></tr>
</table>

Parameter(cont.)	**Description**(cont.)	
<destination>	The destination address of the packets. You can specify a single host, a subnet, or all destinations. The following are the valid formats for specifying the destination:	
	<ip-addr>/<prefix>	An IPv4 address, followed by a forward slash, then the prefix length. This matches any destination IP address within the specified subnet.
	`any`	Matches any destination IP address.
<destport>	The destination port number, specified as an integer between 0 and 65535.	
`eq`	Matches port numbers equal to the port number specified immediately after this parameter.	
`lt`	Matches port numbers less than the port number specified immediately after this parameter.	
`gt`	Matches port numbers greater than the port number specified immediately after this parameter.	
`ne`	Matches port numbers not equal to the port number specified immediately after this parameter.	
`log`	Log the results.	

Mode IPv4 Extended ACL Configuration

Default Any traffic controlled by a software ACL that does not explicitly match a filter is denied.

Usage An ACL can be configured with multiple ACL filters using sequence numbers. If the sequence number is omitted, the next available multiple of 10 will be used as the sequence number for the new filter. A new ACL filter can be inserted into the middle of an existing list by specifying the appropriate sequence number.

Note The access control list being configured is selected by running the access-list (extended numbered) command or the access-list extended (named) command, with the required access control list number, or name - but with no further parameters selected.

Note Software ACLs will **deny** access unless **explicitly permitted** by an ACL action.

Example 1 [creating a list] To add a new entry to the access-list named `my-list` that will reject TCP packets from `10.0.0.1` on TCP port `10` to `192.168.1.1` on TCP port `20`, use the commands:

```
                 awplus# configure terminal
         awplus(config)# access-list extended my-list
awplus(config-ip-ext-acl)# deny tcp 10.0.0.1/32 eq 10
                          192.168.1.1/32 eq 20
```

Example 2 [adding to a list]

To insert a new entry with sequence number 5 of the access-list named `my-list` that will accept UDP packets from `10.1.1.0/24` network to `192.168.1.0/24` network on UDP port `80`, use the commands:

```
              awplus# configure terminal

      awplus(config)# access-list extended my-list

awplus(config-ip-ext-acl)# 5 permit udp 10.1.1.0/24
                        192.168.1.0/24 eq 80
```

Related Commands

access-list extended (named)
access-list (extended numbered)
show running-config
show ip access-list

access-list standard (named)

This command configures a standard named access-list that permits or denies packets from a specific source IP address. You can either create a standard named ACL together with an ACL filter entry in the Global Configuration mode, or you can use the IPv4 Standard ACL Configuration mode for sequenced ACL filter entry after first entering an access-list name.

The **no** variant of this command removes a specified standard named access-list.

Syntax [list-name]

```
access-list standard <standard-access-list-name>

no access-list standard <standard-access-list-name>
```

Parameter	Description
<standard-access-list-name>	Specify a name for the standard access-list.

Syntax [deny|permit]

```
access-list standard <standard-access-list-name> {deny|permit}
    <source>

no access-list standard <standard-access-list-name> {deny|permit}
    <source>
```

Parameter	Description	
<standard-access-list-name>	Specify a name for the standard access-list.	
deny	The access-list rejects packets that match the source filtering specified with this command.	
permit	The access-list permits packets that match the source filtering specified with this command.	
<source>	The source address of the packets. You can specify a single host, a subnet, or all sources. The following are the valid formats for specifying the source:	
	<ip-addr>/<prefix>	An IPv4 address, followed by a forward slash, then the prefix length. This matches any source IP address within the specified subnet.
	any	Matches any source IP address.

Mode Global Configuration

Default Any traffic controlled by a software ACL that does not explicitly match a filter is denied.

Usage Use this command when configuring a standard named access-list for filtering IP software packets. For backwards compatibility you can either create the access-list from within this command, or you can enter this command followed by only the standard access-list name then enter. This latter method moves you to the IPv4 Standard ACL Configuration mode for the selected standard named access-list, and from here you can configure the deny or permit filters for this selected standard named access-list.

See the table "IPv4 Software Access List Commands and Prompts" in this chapter which shows the prompts at which ACL commands are entered. See the relevant links shown for the **Related Commands**.

Note Software ACLs will **deny** access unless **explicitly permitted** by an ACL action.

Examples

To define a standard access-list named `my-list` and deny any packets from any source, use the commands:

```
       awplus#  configure terminal
awplus(config)#  access-list standard my-list deny any
```

Alternatively, to define a standard access-list named `my-list` and enter the IPv4 Standard ACL Configuration mode to deny any packets from any source, use the commands:

```
               awplus#  configure terminal
       awplus(config)#  access-list standard my-list
awplus(config-ip-std-acl)#  5 deny any
```

Related Commands

(access-list standard named filter)
show running-config
show ip access-list

access-list (standard numbered)

This command configures a standard numbered access-list that permits or denies packets from a specific source IP address. You can either create a standard numbered ACL together with an ACL filter entry in the Global Configuration mode, or you can use the IPv4 Standard ACL Configuration mode for sequenced ACL filter entry after first entering an access-list number.

The **no** variant of this command removes a specified standard numbered access-list.

Syntax [list-number]

```
access-list {<1-99>|<1300-1999>}

no access-list {<1-99>|<1300-1999>}
```

Parameter	Description
<1-99>	IP standard access-list.
<1300-1999>	IP standard access-list (expanded range).

Syntax [deny|permit]

```
access-list {<1-99>|<1300-1999>} {deny|permit} <source>
no access-list {<1-99>|<1300-1999>} {deny|permit} <source>
```

Parameter	Description	
<1-99>	IP standard access-list.	
<1300-1999>	IP standard access-list (expanded range).	
`deny`	Access-list rejects packets from the specified source.	
`permit`	Access-list accepts packets from the specified source.	
<source>	The source address of the packets. You can specify a single host, a subnet, or all sources. The following are the valid formats for specifying the source:	
	<ip-addr> *<reverse-mask>*	Enter an IPv4 address followed by a reverse mask in dotted decimal format. For example, entering `192.168.1.1 0.0.0.255` is the same as entering `192.168.1.1/24`.
	`any`	Matches any source IP address.

Mode Global Configuration

Default Any traffic controlled by a software ACL that does not explicitly match a filter is denied.

Usage Use this command when configuring a standard numbered access-list for filtering IP software packets. For backwards compatibility you can either create the access-list from within this command, or you can enter this command followed by only the standard access-list name. This moves you to the IPv4 Standard ACL Configuration mode for the selected standard numbered access-list, and from here you can configure the deny or permit filters for this selected standard numbered access-list.

See the table "IPv4 Software Access List Commands and Prompts" in this chapter which shows the prompts at which ACL commands are entered. See the relevant links shown for the **Related Commands**.

Note Software ACLs will **deny** access unless **explicitly permitted** by an ACL action.

Example To create ACL number 67 that will deny packets from subnet 172.16.10, use the commands:

```
        awplus#  configure terminal
awplus(config)#  access-list 67 deny 172.16.10.0 0.0.0.255
```

Alternatively, to enter the IPv4 Standard ACL Configuration mode to create the ACL filter and deny packets from subnet 172.16.10.0 for the standard numbered access-list 67, use the commands:

```
               awplus#  configure terminal
       awplus(config)#  access-list 67
awplus(config-ip-std-acl)#  deny 172.16.10.0 0.0.0.255
```

Related Commands (access-list standard named filter)
show running-config
show ip access-list

(access-list standard named filter)

This ACL filter adds a source IP address filter entry to a current named standard access-list. If the sequence number is specified, the new filter entry is inserted at the specified location. Otherwise, the new entry is added at the end of the access-list.

The **no** variant of this command removes a source IP address filter entry from the current named standard access-list. You can specify the source IP address filter entry for removal by entering either its sequence number (e.g. `no 10`), or by entering its source IP address filter profile without specifying its sequence number.

Note that the sequence number can be found by running the show access-list (IPv4 Software ACLs) command.

Syntax

```
[<sequence-number>] {deny|permit} {<source> [exact-match]|any}

no {deny|permit} {<source> [exact-match]|any}

no <sequence-number>
```

Parameter	Description	
<sequence-number>	<1-65535> The sequence number for the filter entry of the selected access control list.	
deny	Access-list rejects packets of the source filtering specified.	
permit	Access-list allows packets of the source filtering specified	
<source>	The source address of the packets. You can specify either a subnet or all sources. The following are the valid formats for specifying the source:	
	<ip-addr>/<prefix>	An IPv4 address, followed by a forward slash, then the prefix length. This matches any destination IP address within the specified subnet.
	<ip-addr>	An IPv4 address in a.b.c.d format.
exact-match	Specify an exact IP prefix to match on.	
any	Matches any source IP address.	

Mode IPv4 Standard ACL Configuration

Default Any traffic controlled by a software ACL that does not explicitly match a filter is denied.

Usage An ACL can be configured with multiple ACL filters using sequence numbers. If the sequence number is omitted, the next available multiple of 10 will be used as the sequence number for the new filter. A new ACL filter can be inserted into the middle of an existing list by specifying the appropriate sequence number.

Note The access control list being configured is selected by running the access-list standard (named) command with the required access control list number, or name, but with no further parameters selected.

Note Software ACLs will **deny** access unless **explicitly permitted** by an ACL action.

Example Use the following commands to add a new filter entry to access-list `my-list` that will reject IP address `10.1.1.1`:

```
               awplus# configure terminal
       awplus(config)# access-list standard my-list
awplus(config-ip-std-acl)# deny 10.1.1.1/32
```

Example Use the following commands to insert a new filter entry into access-list `my-list` at sequence position number `15` that will accept IP network `10.1.2.0`:

```
               awplus# configure terminal
       awplus(config)# access-list standard my-list
awplus(config-ip-std-acl)# 15 permit 10.1.2.0/24
```

Related Commands access-list standard (named)
show running-config
show ip access-list

(access-list standard numbered filter)

This ACL filter adds a source IP address filter entry to a current standard numbered access-list. If a sequence number is specified, the new filter entry is inserted at the specified location. Otherwise, the new filter entry is added at the end of the access-list.

The **no** variant of this command removes a source IP address filter entry from the current standard numbered access-list. You can specify the source IP address filter entry for removal by entering either its sequence number (e.g. `no 10`), or by entering its source IP address filter profile without specifying its sequence number.

Note that the sequence number can be found by running the show access-list (IPv4 Software ACLs) command.

Syntax

```
[<sequence-number>] {deny|permit} {<source>|host <host-address>|any}
```

```
no {deny|permit} {<source>|host <host-address>|any}
```

```
no <sequence-number>
```

Parameter	Description	
`<sequence-number>`	`<1-65535>` The sequence number for the filter entry of the selected access control list.	
`deny`	Access-list rejects packets of the type specified.	
`permit`	Access-list allows packets of the type specified	
`<source>`	The source address of the packets. You can specify either a subnet or all sources. The following are the valid formats for specifying the source:	

	`<ip-addr>`
	`<reverse-mask>`
		Enter a reverse mask for the source address in dotted decimal format. For example, entering `192.168.1.1 0.0.0.255` is the same as entering `192.168.1.1/24`.

	`<ip-addr>`	An IPv4 address in a.b.c.d format.
`host`	A single source host.	
`<host-address>`	Single source host address.	
`any`	Matches any source IP address.	

Mode IPv4 Standard ACL Configuration

Default Any traffic controlled by a software ACL that does not explicitly match a filter is denied.

Usage An ACL can be configured with multiple ACL filters using sequence numbers. If the sequence number is omitted, the next available multiple of 10 will be used as the sequence number for the new filter. A new ACL filter can be inserted into the middle of an existing list by specifying the appropriate sequence number.

Note The access control list being configured is selected by running the access-list standard (named) command with the required access control list number, or name, but with no further parameters selected.

Note Software ACLs will **deny** access unless **explicitly permitted** by an ACL action.

Example To add a new entry accepting the IP network `10.1.1.0/24` at the sequence number `15` position, use the commands:

```
                awplus# configure terminal
        awplus(config)# access-list 99
awplus(config-ip-std-acl)# 15 permit 10.1.2.0 0.0.0.255
```

Related Commands access-list (standard numbered)
show running-config
show ip access-list

clear ip prefix-list

Use this command to reset the hit count to zero in the prefix-list entries.

Syntax

```
clear ip prefix-list [<list-name>] [<ip-address>/<mask>]
```

Parameter	Description
<list-name>	The name of the prefix-list.
<ip-address>/<mask>	The IP prefix and length.

Mode Privileged Exec

Example To clear a prefix-list named List1:

```
awplus# clear ip prefix-list List1
```

dos

Use this command to configure Denial-of-Service (DoS) features for a port. Six different DoS attacks can be detected: IP Options, Land, Ping-of-Death, Smurf, Synflood and Teardrop.

When the attack is detected, three different actions are available:

1. Shutdown the port for one minute
2. Cause an SNMP trap.
3. Send traffic to the mirror port

Syntax

```
dos {ipoptions|land|ping-of-death|smurf broadcast <ip-address>|
    synflood|teardrop} action {shutdown|trap|mirror}
```

Parameter	Description
`dos`	Denial-Of-Service.
`ipoptions`	IP Options attack.
`land`	Land attack.
`ping-of-death`	Large ping attack.
`smurf`	Ping to broadcast address.
`broadcast`	Broadcast.
`<ip-address>`	Local IP Broadcast Address.
`synflood`	SYN flood attack.
`teardrop`	IP fragmentation attack.
`action`	Action.
`shutdown`	Shutdown port.
`trap`	Trap to SNMP.
`mirror`	Send packets to mirror port.

Mode Interface Configuration for a switch port interface.

Default DoS attack detection is not configured by default on any switch port interface.

Usage See the below table for more information about the DoS attacks recognized by this command:

Type of DoS attack	Description
`ipoptions`	This type of attack occurs when an attacker sends packets containing bad IP options to a victim node. There are many different types of IP options attacks and this software does not try to distinguish between them. Rather, if this defence is activated, the number of ingress IP packets containing IP options is counted. If the number exceeds 20 packets per second, the switch considers this a possible IP options attack. This defence does not require the CPU to monitor packets, so does not put extra load on the switch's CPU.
`land`	This type of attack occurs when the Source IP and Destination IP address are the same. This can cause a target host to be confused. Since packets with the same source and destination addresses should never occur, these packets are dropped when this attack is enabled. This defence does not require the CPU to monitor packets, so does not put extra load on the switch's CPU.
`ping-of-death`	This type of attack results from a fragmented packet which, when reassembled, would exceed the maximum size of a valid IP datagram. To detect this attack, the final fragment of ICMP packets has to be sent to the CPU for inspection. This defence can therefore load the CPU. Note that the extra CPU load will not affect normal traffic switching between ports, but other protocols such as IGMP and STP may be affected. This defence is not recommended where a large number of fragmented packets are expected.
`smurf`	This type of attack is an ICMP ping packet to a broadcast address. Although routers should not forward packets to local broadcast addresses anymore (see RFC2644), the Smurf attack can still be explicitly discarded with this command. In order for the Smurf attack to work, the broadcast IP address is required. Any ICMP Ping packet with this destination address is considered an attack. This defence does not require the CPU to monitor packets, so does not put extra load on the switch's CPU.
`synflood`	In this type of attack, an attacker, seeking to overwhelm a victim with TCP connection requests, sends a large number of TCP SYN packets with bogus source addresses to the victim. The victim responds with SYN ACK packets, but since the original source addresses are bogus, the victim node does not receive any replies. If the attacker sends enough requests in a short enough period, the victim may freeze operations once the requests exceed the capacity of its connections queue. To defend against this form of attack, a switch port monitors the number of ingress TCP-SYN packets it receives. An attack is recorded if a port receives more 60 TCP-SYN packets per second.
`teardrop`	In this DoS attack, an attacker sends a packet in several fragments with a bogus offset value, used to reconstruct the packet, in one of the fragments to a victim. This results in the victim being unable to reassemble the packet, possibly causing it to freeze operations.

Examples To configure **smurf** DoS detection on `port1.0.1`, and shutdown the interface if an attack is detected, use the commands:

```
awplus#           configure terminal
awplus(config)#   interface port1.0.1
awplus(config-if)# dos smurf broadcast 192.168.1.0 action
                   shutdown
```

To configure **land** DoS detection on `port1.0.1`, and shutdown the interface if an attack is detected, use the commands:

```
awplus#           configure terminal
awplus(config)#   interface port1.0.1
awplus(config-if)# dos land action shutdown
```

To configure **ipoptions** DoS detection on `port1.0.1`, and shutdown the interface if an attack is detected, use the commands:

```
awplus#           configure terminal
awplus(config)#   interface port1.0.1
awplus(config-if)# dos ipoptions action shutdown
```

To configure **ping-of-death** DoS detection on `port1.0.1`, and shutdown the interface if an attack is detected, use the commands:

```
awplus#           configure terminal
awplus(config)#   interface port1.0.1
awplus(config-if)# dos ping-of-death action shutdown
```

To configure **synflood** DoS detection on `port1.0.1`, and shutdown the interface if an attack is detected, use the commands:

```
awplus#           configure terminal
awplus(config)#   interface port1.0.1
awplus(config-if)# dos synflood action shutdown
```

To configure **teardrop** DoS detection on `port1.0.1`, and shutdown the interface if an attack is detected, use the commands:

```
awplus#           configure terminal
awplus(config)#   interface port1.0.1
awplus(config-if)# dos teardrop action shutdown
```

Related Commands show dos interface

ip prefix-list

Use this command to create an entry for a prefix list.

Use the **no** variant of this command to delete the prefix-list entry.

Syntax

```
ip prefix-list <list-name> seq <1-429496725>
   {deny|permit}
   {any|<ip-prefix>}
   [ge <0-32>] [le <0-32>]
```

```
ip prefix-list <list-name> description <text>
```

```
ip prefix-list sequence-number
```

```
no ip prefix-list <list-name> seq <1-429496725>
```

```
no ip prefix-list <list-name> description <text>
```

```
no ip prefix-list sequence-number
```

Parameter	Description
`<list-name>`	Specifies the name of a prefix list.
`seq <1-429496725>`	Sequence number of the prefix list entry.
`deny`	Specifies that the prefixes are excluded from the list.
`permit`	Specifies that the prefixes are included in the list.
`<ip-prefix>`	A.B.C.D/M specifies the IP address and length of the network mask.
`any`	Any prefix match. Same as **0.0.0.0/0 le 32**.
`le <0-32>`	Specifies the maximum prefix length to be matched.
`ge <0-32>`	Specifies the minimum prefix length to be matched.
`description <text>`	Text description of the prefix list.
`sequence-number`	Specify sequence numbers included or excluded in prefix list.

Mode Global Configuration

Usage When the device processes a prefix list, it starts to match prefixes from the top of the prefix list, and stops whenever a match or deny occurs. To promote efficiency, use the **seq** parameter and place common matches or denials towards the top of the list. If you do not use the **seq** parameter, the sequence values are generated in the sequence of 5.

The parameters **ge** and **le** specify the range of the prefix lengths to be matched. When setting these parameters, set the **le** value to be less than 32, and the **ge** value to be less than the **le** value.

Related Commands match ip address
match route-type

maximum-access-list

Sets the maximum number of filters that can be added to any access-list. These are access-lists within the ranges <1-199>, <1300-1999> and <2000-2699> and named standard and extended access-lists.

The **no** variant of this command removes the limit on the number of filters that can be added to a software access-list

Syntax

```
maximum-access-list <1-4294967294>

no maximum-access-list
```

Parameter	Description
<1-4294967294>	Filter range.

Mode Global Configuration

Example To set the maximum number of software filters to 200:

```
        awplus# configure terminal
awplus(config)# maximum-access-list 200
```

Related Commands remote-command <1-4> show

show access-list (IPv4 Software ACLs)

Use this command to display the specified access-list, or all access-lists if none have been specified. Note that only defined access-lists are displayed. An error message is displayed for an undefined access-list

Syntax

```
show access-list
    [<1-99>|<100-199>|<1300-1999>|<2000-2699>|<3000-3699>|
    <4000-4499>|<access-list-name>]
```

Parameter	Description
<1-99>	IP standard access-list.
<100-199>	IP extended access-list.
<1300-1999>	IP standard access-list (standard - expanded range).
<2000-2699>	IP extended access-list (extended - expanded range).
<3000-3699>	Hardware IP access-list.
<4000-4499>	Hardware MAC access-list.
<access-list-name>	IP named access-list.

Mode User Exec and Privileged Exec

Example To show all access-lists configured on the switch:

```
awplus# show access-list
```

```
Standard IP access list 1
    deny 172.16.2.0, wildcard bits 0.0.0.255
Standard IP access list 20
    deny 192.168.10.0, wildcard bits 0.0.0.255
    deny 192.168.12.0, wildcard bits 0.0.0.255
Hardware IP access list 3001
    permit ip 192.168.20.0 255.255.255.0 any
Hardware IP access list 3020
    permit tcp any 192.0.2.0/24
awplus#show access-list 20
```

Example To show the access-list with an ID of 20:

```
awplus# show access-list 20
```

```
Standard IP access-list 20
    deny 192.168.10.0, wildcard bits 0.0.0.255
    deny 192.168.12.0, wildcard bits 0.0.0.255
```

Note the below error message if you attempt to show an undefined access-list:

```
awplus# show access-list 2
```

```
% Can't find access-list 2
```

Related Commands access-list standard (named)
access-list (standard numbered)
access-list (extended numbered)

show dos interface

Use this command to display the Denial-of-Service (DoS) features configured on a switch port interface from the dos command. See the dos command for descriptions of DoS attack types.

Syntax `show dos interface {<port-list>}`

Parameter	Description
<port-list>	Specify the switch port or port list to display DoS configuration options set with the dos command.

Mode User Exec and Privileged Exec

Output Figure 33-1: Example output from the **show dos interface** command prior to a DoS attack

```
awplus#configure terminal
Enter configuration commands, one per line. End with CTNTL/Z.
awplus(config)#interface port1.0.1
awplus(config-if)#dos synflood action shutdown
awplus(config-if)#exit
awplus(config)#exit
awplus#show dos interface port1.0.1

DoS settings  for interface port1.0.1
-----------------------------------------
Port status          : Enabled
ipoptions            : Disabled
land                 : Disabled
ping-of-death        : Disabled
smurf                : Disabled
synflood             : Enabled
   Action            : Shutdown port
   Attacks detected : 0
teardrop             : Disabled
awplus#
```

Figure 33-2: Example output from the **show dos interface** command after a **synflood** DoS attack:

```
awplus#show dos interface port1.0.1

DoS settings for interface port1.0.1
-----------------------------------------
Port status          : Enabled
ipoptions            : Disabled
land                 : Disabled
ping-of-death        : Disabled
smurf                : Disabled
synflood             : Enabled
   Action            : Shutdown port
   Attacks detected : 1
teardrop             : Disabled
awplus#
```

Table 33-5: Parameters in the **show dos interface** command output:

Type of DoS attack	Description
`Port status`	Displays `Enabled` when the port is configured as being administratively up after issuing the **no shutdown** command. Displays `Disabled` when the port is configured as being administratively down with the **shutdown** command.
`ipoptions`	Displays `Enabled` when the **ipoptions** parameter is configured with the dos command, plus the action (`Shutdown port`, `Mirror port`, or `Trap port`) and the number of instances of any **ipoptions** DoS attacks that have occurred on the interface. Displays `Disabled` when the **ipoptions** parameter is not configured with the dos command.
`land`	Displays `Enabled` when the **land** parameter is configured with the dos command, plus the action (`Shutdown port`, `Mirror port`, or `Trap port`) and the number of instances of any **land** DoS attacks that have occurred on the interface. Displays `Disabled` when the **land** parameter is not configured with the dos command.
`ping-of-death`	Displays `Enabled` when the **ping-of-death** parameter is configured with the dos command, plus the action (`Shutdown port`, `Mirror port`, or `Trap port`) and the number of instances of any **ping-of-death** DoS attacks that have occurred on the interface. Displays `Disabled` when the **ping-of-death** parameter is not configured with the dos command.
`smurf`	Displays `Enabled` when the **smurf** parameter is configured with the dos command, plus the action (`Shutdown port`, `Mirror port`, or `Trap port`) and the number of instances of any **smurf** DoS attacks that have occurred on the interface. Displays `Disabled` when the **smurf** parameter is not configured with the dos command.
`synflood`	Displays `Enabled` when the **synflood** parameter is configured with the dos command, plus the action (`Shutdown port`, `Mirror port`, or `Trap port`) and the number of instances of any **synflood** DoS attacks that have occurred on the interface. Displays `Disabled` when the **synflood** parameter is not configured with the dos command.
`teardrop`	Displays `Enabled` when the **teardrop** parameter is configured with the dos command, plus the action (`Shutdown port`, `Mirror port`, or `Trap port`) and the number of instances of any **teardrop** DoS attacks that have occurred on the interface. Displays `Disabled` when the **teardrop** parameter is not configured with the dos command.

Related Commands dos

show ip access-list

Use this command to display IP access-lists.

Syntax

```
show ip access-list [<1-99>|<100-199>|<1300-1999>|<2000-2699>|
    <access-list-name>]
```

Parameter	Description
<1-99>	IP standard access-list.
<100-199>	IP extended access-list.
<1300-1999>	IP standard access-list (expanded range).
<2000-2699>	IP extended access-list (expanded range).
<access-list-name>	IP named access-list.

Mode User Exec and Privileged Exec

Example

```
awplus# show ip access-list
```

Output Figure 33-3: Example output from the **show ip access-list** command

```
Standard IP access-list 1
    permit 172.168.6.0, wildcard bits 0.0.0.255
    permit 192.168.6.0, wildcard bits 0.0.0.255
```

Chapter 34: IPv6 Software Access Control List (ACL) Commands

Introduction

This chapter provides an alphabetical reference for the IPv6 Software Access Control List (ACL) commands, and contains detailed command information and command examples about IPv6 software ACLs as applied to Routing and Multicasting, which are not applied to interfaces.

Note See Chapter 31, Access Control Lists Introduction for descriptions of ACLs, and for further information about rules when applying ACLs see the ACL Rules section.

See ACL Filter Sequence Numbers and ACL Filter Sequence Number Behavior sections in Chapter 31, Access Control Lists Introduction about ACL Filters.

See all relevant Routing commands and configurations in "Layer Three, Switching and Routing" and all relevant Multicast commands and configurations in "Multicast Applications".

To apply ACLs to an LACP channel group, apply it to all the individual switch ports in the channel group. To apply ACLs to a static channel group, apply it to the static channel group itself. For more information on link aggregation see Chapter 20, Link Aggregation Introduction and Configuration, and Chapter 21, Link Aggregation Commands.

Note that text in parenthesis in command names indicates usage not keyword entry. For example, **ipv6-access-list (named)** indicates named IPv6 ACLs entered as `ipv6-access-list <name>` where *<name>* is a placeholder not a keyword.

Note also that parenthesis surrounding ACL filters indicates the type of ACL filter not the keyword entry in the CLI, such as **(ipv6 access-list standard IPv6 filter)** represents command entry in the format shown in the syntax `[<sequence-number>] {deny|permit} {<IPv6-source-address/prefix-length>|any}`.

Note Software ACLs will **deny** access unless **explicitly permitted** by an ACL action.

IPv6 Software Access List Commands and Prompts

Many of the ACL commands operate from sub-modes that are specific to particular ACL types. The table "IPv6 Software Access List Commands and Prompts" shows the CLI prompts at which ACL commands are entered.

Table 34-1: IPv6 Software Access List Commands and Prompts

Command Name	Command Mode	Prompt
show ipv6 access-list (IPv6 Software ACLs)	Privileged Exec	`awplus#`
ipv6 access-list extended (named)	Global Configuration	`awplus(config)#`
ipv6 access-list standard (named)	Global Configuration	`awplus(config)#`
(ipv6 access-list extended IP protocol filter)	IPv6 Extended ACL Configuration	`awplus(config-ipv6-ext-acl)#`
(ipv6 access-list extended TCP UDP filter)	IPv6 Extended ACL Configuration	`awplus(config-ipv6-ext-acl)#`
(ipv6 access-list standard filter)	IPv6 Standard ACL Configuration	`awplus(config-ipv6-std-acl)#`

Command List

ipv6 access-list extended (named)

Use this command when configuring an IPv6 extended access-list for filtering frames that permit or deny IP, ICMP, TCP, UDP packets or ICMP packets with a specific value based on the source or destination.

The **no** variant of this command removes a specified IPv6 extended access-list.

Syntax [list-name]

```
ipv6 access-list extended <list-name>

no ipv6 access-list extended <list-name>
```

Parameter	Description
<list-name>	A user-defined name for the IPv6 software extended access-list.

Syntax [any|icmp|ip]

```
ipv6 access-list extended <list-name>
    {deny|permit} {any|icmp|ip}
    {<ipv6-source-address/prefix-length>|any}
    {<ipv6-destination-address/prefix-length>|any}
    [<icmp-type <icmp-type>][log]

no ipv6 access-list extended <list-name>
    {deny|permit} {any|icmp|ip}
    {<ipv6-source-address/prefix-length>|any}
    {<ipv6-destination-address/prefix-length>|any}
    [<icmp-type <icmp-type>][log]
```

Syntax [tcp|udp]

```
ipv6 access-list extended <list-name>
    {deny|permit} {tcp|udp}
    {<ipv6-source-address/prefix-length>|any}
    {eq <sourceport>|lt <sourceport>|gt <sourceport>|ne <sourceport>}
    {<ipv6-destination-address/prefix-length>|any}
    {eq <destport>|lt <destport>|gt <destport>|ne <destport>}
    [log]

no ipv6 access-list extended <list-name> {deny|permit} {tcp|udp}
    {<ipv6-source-address/prefix-length>|any}
    {eq <sourceport>|lt <sourceport>|gt <sourceport>|ne <sourceport>}
    {<ipv6-destination-addr/prefix-length>|any}
    {eq <destport>|lt <destport>|gt <destport>|ne <destport>}
    [log]
```

Parameter	Description
<list-name>	A user-defined name for the IPv6 software extended access-list.
deny	The IPv6 software extended access-list rejects packets that match the type, source, and destination filtering specified with this command.
permit	The IPv6 software extended access-list permits packets that match the type, source, and destination filtering specified with this command.

Parameter(cont.)	**Description**(cont.)
`any`	For ICMP|IP The IPv6 software extended access-list matches any type of packet.
`ip`	For ICMP|IP The IPv6 software extended access-list matches only IP packets.
`icmp`	For ICMP|IP The IPv6 software extended access-list matches only ICMP packets.
`tcp`	For TCP/UDP The IPv6 software extended access-list matches only TCP packets.
`udp`	For TCP/UDP The IPv6 software extended access-list matches only UDP packets.
`<ipv6-source-address/prefix-length>`	Specifies a source address and prefix length. The IPv6 address prefix uses the format X:X::/prefix-length. The prefix-length is usually set between 0 and 64, or has the value 128.
`<ipv6-destination-address/prefix-length>`	Specifies a destination address and prefix length. The IPv6 address uses the format X:X::X:X/Prefix-Length. The prefix-length is usually set between 0 and 64, or has the value 128.
`any`	Matches any IPv6 address.
`<sourceport>`	For TCP/UDP The source port number, specified as an integer between 0 and 65535.
`<destport>`	For TCP/UDP The destination port number, specified as an integer between 0 and 65535.
`icmp-type`	For ICMP|IP Matches only a specified type of ICMP messages. This is valid only when the filtering is set to match ICMP packets.
`eq`	For TCP/UDP Matches port numbers equal to the port number specified immediately after this parameter.
`lt`	For TCP/UDP Matches port numbers less than the port number specified immediately after this parameter.
`gt`	For TCP/UDP Matches port numbers greater than the port number specified immediately after this parameter.
`ne`	For TCP/UDP Matches port numbers not equal to the port number specified immediately after this parameter.

<table>
<tr><th>Parameter(cont.)</th><th colspan="2">Description(cont.)</th></tr>
<tr><td rowspan="15"><i>&lt;icmp-type&gt;</i></td><td colspan="2">For ICMP|IP
The ICMP type, as defined in RFC792 and RFC950. Specify one of the following integers to create a filter for the ICMP message type:</td></tr>
<tr><td>0</td><td>Echo replies.</td></tr>
<tr><td>3</td><td>Destination unreachable messages.</td></tr>
<tr><td>4</td><td>Source quench messages.</td></tr>
<tr><td>5</td><td>Redirect (change route) messages.</td></tr>
<tr><td>8</td><td>Echo requests.</td></tr>
<tr><td>11</td><td>Time exceeded messages.</td></tr>
<tr><td>12</td><td>Parameter problem messages.</td></tr>
<tr><td>13</td><td>Timestamp requests.</td></tr>
<tr><td>14</td><td>Timestamp replies.</td></tr>
<tr><td>15</td><td>Information requests.</td></tr>
<tr><td>16</td><td>Information replies.</td></tr>
<tr><td>17</td><td>Address mask requests.</td></tr>
<tr><td>18</td><td>Address mask replies.</td></tr>
<tr><td colspan="2"></td></tr>
<tr><td><code>log</code></td><td colspan="2">Logs the results.</td></tr>
</table>

Mode Global Configuration

Default Any traffic controlled by a software ACL that does not explicitly match a filter is denied.

Usage Use IPv6 extended access-lists to control the transmission of IPv6 packets on an interface, and restrict the content of routing updates. The switch stops checking the IPv6 extended access-list when a match is encountered.

For backwards compatibility you can either create IPv6 extended access-lists from within this command, or you can enter `ipv6 access-list extended` followed by only the IPv6 extended access-list name. This latter (and preferred) method moves you to the `(config-ipv6-ext-acl)` prompt for the selected IPv6 extended access-list number, and from here you can configure the filters for this selected access-list.

Note Software ACLs will **deny** access unless **explicitly permitted** by an ACL action.

Example 1 [creating a list]

To add a new filter to the access-list named `my-list` that will reject incoming ICMP packets from `2001:0db8::0/64` to `2001:0db8::f/64`, use the commands:

```
                 awplus# configure terminal
          awplus(config)# ipv6 access-list extended my-list
awplus(config-ipv6-ext-acl)# icmp 2001:0db8::0/64 2001:0db8::f/64
```

Example 2 [adding to a list]

To insert a new filter at sequence number 5 of the access-list named `my-list` that will accept ICMP type 8 packets from the `2001:0db8::0/64` network to the `2001:0db8::f/64` network, use the commands:

```
                 awplus# configure terminal
          awplus(config)# ipv6 access-list extended my-list
awplus(config-ipv6-ext-acl)# 5 icmp 2001:0db8::0/64
                            2001:0db8::f/64
```

Example 3 [list with filter]

To create the access-list named `TK` to deny TCP protocols, use the commands:

```
       awplus# configure terminal
awplus(config)# ipv6 access-list extended TK deny tcp any eq
                14 any lt 12 log
```

Related Commands

ipv6 access-list extended proto
(ipv6 access-list extended IP protocol filter)
(ipv6 access-list extended TCP UDP filter)
show ipv6 access-list (IPv6 Software ACLs)
show running-config

ipv6 access-list extended proto

Use this command when configuring an IPv6 extended access-list for filtering frames that permit or deny packets with a specific value based on the IP protocol number specified.

The **no** variant of this command removes a specified IPv6 extended access-list with an IP protocol number.

Syntax

```
ipv6 access-list extended <list-name>
    {deny|permit} proto <ip-protocol>}
    {<ipv6-source-address/prefix>|any}
    {<ipv6-destination-address/prefix>|any} [log]

no ipv6 access-list extended <list-name>
    {deny|permit} proto <ip-protocol>
    {<ipv6-source-address/prefix>|any}
    {<ipv6-destination-address/prefix>|any} [log]
```

Parameter	Description
<list-name>	A user-defined name for the IPv6 software extended access-list.
`deny`	Specifies the packets to reject.
`permit`	Specifies the packets to accept.
`proto`	The IP Protocol type specified by it protocol number <1-255>.
<ip-protocol>	The IP protocol number, as defined by IANA (Internet Assigned Numbers Authority http://www.iana.org/assignments/protocol-numbers). **Protocol Number** 1 2 3 4 5 6 8

Parameter(cont.)	Description(cont.)
<ip-protocol> (cont.)	***Protocol Number*** 9
	11
	17
	20
	27
	28
	29
	30
	33
	48
	50
	51
	54
	58
	59
	60
	88
	89
	97
	98
	108
	112
	134
	135
	136
	137
	138
	139-252
	253
	254
	255
<ipv6-source-address/prefix>	IPv6 source address, or local address. The IPv6 address uses the format X:X::X:X/Prefix-Length. The prefix-length is usually set between 0 and 64, or has the value 128.
any	Any source address or local address.

Parameter(cont.)	Description(cont.)
<ipv6-destination-address/prefix>	IPv6 destination address, or local address. The IPv6 address uses the format X:X::X:X/Prefix-Length. The prefix-length is usually set between 0 and 64, or has the value 128.
`any`	Any destination address or remote address.
`log`	Log the results.

Mode Global Configuration

Default Any traffic controlled by a software ACL that does not explicitly match a filter is denied.

Usage Use IPv6 extended access-lists to control the transmission of IPv6 packets on an interface, and restrict the content of routing updates. The switch stops checking the IPv6 extended access-list when a match is encountered.

The filter entry will match on any IP protocol type packet that has the specified source and destination IPv6 addresses and the specified IP protocol type. The parameter `any` may be specified if an address does not matter.

Note Software ACLs will **deny** access unless **explicitly permitted** by an ACL action.

Examples To create the IPv6 access-list named `ACL-1` to deny IP protocol 9 packets from `2001:0db8:1::1/128` to `2001:0db8:f::1/128`, use the commands:

```
        awplus# configure terminal

awplus(config)# ipv6 access-list extended ACL-1 deny proto 9
                2001:0db8:1::1/128 2001:0db8:f::1/128
```

To remove the IPv6 access-list named `ACL-1` to deny IP protocol 9 packets from `2001:0db8:1::1/128` to `2001:0db8:f::1/128`, use the commands:

```
        awplus# configure terminal

awplus(config)# no ipv6 access-list extended ACL-1 deny proto
                10 2001:0db8:1::1/128 2001:0db8:f::1/128
```

Related Commands ipv6 access-list extended (named)
(ipv6 access-list extended IP protocol filter)
show ipv6 access-list (IPv6 Software ACLs)
show running-config

(ipv6 access-list extended IP protocol filter)

Use this ACL filter to add a filter entry for an IPv6 source and destination address and prefix, with or without an IP protocol specified, to the current extended IPv6 access-list. If a sequence is specified, the new entry is inserted at the specified location. Otherwise, the new entry is added at the end of the access-list.

The **no** variant of this command removes a filter entry for an IPv6 source and destination address and prefix, with or without an IP protocol filter entry, from the current extended IPv6 access-list. You can specify the ACL filter entry by entering either its sequence number, or its filter entry profile.

Syntax [ip|proto]

```
[<sequence-number>]
    {deny|permit} {ip|any|proto <ip-protocol>}
    {<ipv6-source-address/prefix>|any}
    {<ipv6-destination-address/prefix>|any} [log]

no {deny|permit} {ip|any|proto <ip-protocol>}
    {<ipv6-source-address/prefix>|any}
    {<ipv6-destination-address/prefix>|any} [log]

no [<sequence-number>]
```

Parameter	Description
<sequence-number>	<1-65535> The sequence number for the filter entry of the selected access control list.
deny	Specifies the packets to reject.
permit	Specifies the packets to accept.
ip	IP packet.
any	Any packet.
proto *<ip-protocol>*	The IP Protocol type specified by it protocol number <1-255>.
<ip-protocol>	The IP protocol number, as defined by IANA (Internet Assigned Numbers Authority http://www.iana.org/assignments/protocol-numbers).
	Protocol Number
	1
	2
	3
	4
	5
	6
	8

Parameter(cont.)	Description(cont.)
<ip-protocol> (cont.)	***Protocol Number*** 9
	11
	17
	20
	27
	28
	29
	30
	33
	48
	50
	51
	54
	58
	59
	60
	88
	89
	97
	98
	108
	112
	134
	135
	136
	137
	138
	139-252
	253
	254
	255
<ipv6-source-address/prefix>	IPv6 source address, or local address. The IPv6 address uses the format X:X::X:X/Prefix-Length. The prefix-length is usually set between 0 and 64, or has the value 128.
any	Any source address or local address.

Parameter(cont.)	Description(cont.)
<ipv6-destination-address/prefix>	IPv6 destination address, or local address. The IPv6 address uses the format X:X::X:X/Prefix-Length. The prefix-length is usually set between 0 and 64, or has the value 128.
`any`	Any destination address or remote address.
`log`	Log the results.

Mode IPv6 Extended ACL Configuration

Default Any traffic controlled by a software ACL that does not explicitly match a filter is denied.

Usage The filter entry will match on any IP protocol type packet that has the specified source and destination IPv6 addresses and the specified IP protocol type. The parameter `any` may be specified if an address does not matter.

Note Software ACLs will **deny** access unless **explicitly permitted** by an ACL action.

Examples To add a new ACL filter entry to the extended IPv6 access-list named `my-list` with sequence number 5 rejecting the IPv6 packet from `2001:db8:1:1` to `2001:db8:f:1`, use the commands:

```
                awplus#  configure terminal
        awplus(config)#  ipv6 access-list extended my-list
awplus(config-ipv6-ext-acl)#  5 deny ip 2001:db8:1::1/128
                              2001:db8:f::1/128
```

To remove the ACL filter entry to the extended IPv6 access-list named `my-list` with sequence number 5, use the commands:

```
                awplus#  configure terminal
        awplus(config)#  ipv6 access-list extended my-list
awplus(config-ipv6-ext-acl)#  no 5
```

Related Commands ipv6 access-list extended (named)
show ipv6 access-list (IPv6 Software ACLs)
show running-config

(ipv6 access-list extended TCP UDP filter)

Use this ACL filter to add a filter entry for an IPv6 source and destination address and prefix, with a TCP (Transmission Control Protocol) or UDP (User Datagram Protocol) source and destination port specified, to the current extended IPv6 access-list. If a sequence number is specified, the new entry is inserted at the specified location. Otherwise, the new entry is added at the end of the access-list.

The **no** variant of this command removes a filter entry for an IPv6 source and destination address and prefix, with a TCP or UDP source and destination port specified, from the current extended IPv6 access-list. You can specify the filter entry for removal by entering either its sequence number, or its filter entry profile.

Syntax [tcp|udp]

```
[<sequence-number>] {deny|permit} {tcp|udp}
   {<ipv6-source-address/prefix>|any}
   {eq <sourceport>|lt <sourceport>|gt <sourceport>|ne <sourceport>}
   {<IPv6-destination-address/prefix>|any}
   {eq <destport>|lt <destport>|gt <destport>|ne <destport>} [log]

no {deny|permit} {tcp|udp}
   {<ipv6-source-address/prefix>|any}
   {eq <sourceport>|lt <sourceport>|gt <sourceport>|ne <sourceport>}}
   {<IPv6-destination-address/prefix>|any}
   {eq <destport>|lt <destport>|gt <destport>|ne <destport>} [log]

no <sequence-number>
```

Parameter	Description
`<sequence-number>`	`<1-65535>` The sequence number for the filter entry of the selected access control list.
`deny`	Specifies the packets to reject.
`permit`	Specifies the packets to accept.
`tcp`	TCP packet.
`udp`	UDP packet.
`<ipv6-source-address/prefix>`	IPv6 source address, or local address. The IPv6 address uses the format X:X::X:X/Prefix-Length. The prefix-length is usually set between 0 and 64, or has the value 128.
`any`	Any source address or local address.
`eq`	Equal to.
`lt`	Less than.
`gt`	Greater than.
`ne`	Not equal to.
`<sourceport>`	The source port number, specified as an integer between 0 and 65535.

Parameter(cont.)	Description(cont.)
<ipv6-destination-address/prefix>	IPv6 destination address, or local address. The IPv6 address uses the format X:X::X:X/Prefix-Length. The prefix-length is usually set between 0 and 64, or has the value 128.
<destport>	The destination port number, specified as an integer between 0 and 65535.
`log`	Log the results.

Mode IPv6 Extended ACL Configuration

Default Any traffic controlled by a software ACL that does not explicitly match a filter is denied.

Usage The filter entry will match on any packet that has the specified source and destination IPv6 addresses and the specified TCP or UDP source and destination port. The parameter `any` may be specified if an address does not matter.

Note Software ACLs will **deny** access unless **explicitly permitted** by an ACL action.

Examples To add a new filter entry with sequence number 5 to the access-list named `my-list` to reject TCP packets from `2001:0db8::0/64` port `10` to `2001:0db8::f/64` port `20`, use the following commands:

```
                awplus# configure terminal
        awplus(config)# ipv6 access-list extended my-list
awplus(config-ipv6-ext-acl)# 5 deny tcp 2001:0db8::0/64 eq 10
                            2001:0db8::f/64 eq 20
```

To add a new filter entry with sequence number 5 to the extended IPv6 access-list named `my-list` to reject UDP packets from `2001:0db8::0/64` port `10` to `2001:0db8::f/64` port `20`, use the following commands:

```
                awplus# configure terminal
        awplus(config)# ipv6 access-list extended my-list
awplus(config-ipv6-ext-acl)# 5 deny udp 2001:0db8::0/64 eq 10
                            2001:0db8::f/64 eq 20
```

To remove the filter entry with sequence number 5 to the extended IPv6 access-list named `my-list`, use the commands:

```
                awplus# configure terminal
        awplus(config)# ipv6 access-list extended my-list
awplus(config-ipv6-ext-acl)# no 5
```

Related Commands ipv6 access-list extended (named)
show ipv6 access-list (IPv6 Software ACLs)
show running-config

ipv6 access-list standard (named)

This command configures an IPv6 standard access-list for filtering frames that permit or deny IPv6 packets from a specific source IPv6 address.

The **no** variant of this command removes a specified IPv6 standard access-list.

Syntax [list-name]

```
ipv6 access-list standard <ipv6-acl-list-name>

no ipv6 access-list standard <ipv6-acl-list-name>
```

Parameter	Description
<ipv6-acl-list-name>	A user-defined name for the IPv6 software standard access-list.

Syntax [deny|permit]

```
ipv6 access-list standard <ipv6-acl-list-name>
    [{deny|permit}
    {<ipv6-source-address/prefix-length>|any}
    [exact-match]]

no ipv6 access-list standard <ipv6-acl-list-name>
    [{deny|permit}
    {<ipv6-source-address/prefix-length>|any}
    [exact-match]]
```

Parameter	Description
<ipv6-acl-list-name>	A user-defined name for the IPv6 software standard access-list.
`deny`	The IPv6 software standard access-list rejects packets that match the type, source, and destination filtering specified with this command.
`permit`	The IPv6 software standard access-list permits packets that match the type, source, and destination filtering specified with this command.
<ipv6-source-address/ prefix-length>	Specifies a source address and prefix length. The IPv6 address prefix uses the format X:X::/prefix-length. The prefix-length is usually set between 0 and 64, or has the value 128.
`any`	Matches any source IPv6 address.
`exact-match`	Exact match of the prefixes.

Mode Global Configuration

Default Any traffic controlled by a software ACL that does not explicitly match a filter is denied.

Usage Use IPv6 standard access-lists to control the transmission of IPv6 packets on an interface, and restrict the content of routing updates. The switch stops checking the IPv6 standard access-list when a match is encountered.

For backwards compatibility you can either create IPv6 standard access-lists from within this command, or you can enter `ipv6 access-list standard` followed by only the IPv6 standard access-list name. This latter (and preferred) method moves you to the `(config-ipv6-std-acl)` prompt for the selected IPv6 standard access-list, and from here you can configure the filters for this selected IPv6 standard access-list.

Note Software ACLs will **deny** access unless **explicitly permitted** by an ACL action.

Example To enter the IPv6 Standard ACL Configuration mode for the access-list named `my-list`, use the commands:

```
              awplus# configure terminal
      awplus(config)# ipv6 access-list standard my-list
awplus(config-ipv6-std-acl)#
```

Related Commands (ipv6 access-list standard filter)
show ipv6 access-list (IPv6 Software ACLs)
show running-config

(ipv6 access-list standard filter)

Use this ACL filter to add a filter entry for an IPv6 source address and prefix length to the current standard IPv6 access-list. If a sequence number is specified, the new entry is inserted at the specified location. Otherwise, the new entry is added at the end of the access-list.

The **no** variant of this command removes a filter entry for an IPv6 source address and prefix from the current standard IPv6 access-list. You can specify the filter entry for removal by entering either its sequence number, or its filter entry profile.

Syntax [icmp]

```
[<sequence-number>] {deny|permit}
    {<ipv6-source-address/prefix-length>|any}

no {deny|permit}
    {<ipv6-source-address/prefix-length>|any}

no <sequence-number>
```

Parameter	Description
`<sequence-number>`	`<1-65535>` The sequence number for the filter entry of the selected access control list.
`deny`	Specifies the packets to reject.
`permit`	Specifies the packets to accept.
`<ipv6-source-address/prefix-length>`	IPv6 source address and prefix-length in the form X:X::X:X/P.
`any`	Any IPv6 source host address.

Mode IPv6 Standard ACL Configuration

Default Any traffic controlled by a software ACL that does not explicitly match a filter is denied.

Usage The filter entry will match on any IPv6 packet that has the specified IPv6 source address and prefix length. The parameter `any` may be specified if an address does not matter.

Note Software ACLs will **deny** access unless **explicitly permitted** by an ACL action.

Examples To add an ACL filter entry with sequence number 5 that will deny any IPv6 packets to the standard IPv6 access-list named `my-list`, enter the commands:

```
               awplus# configure terminal
       awplus(config)# ipv6 access-list standard my-list
awplus(config-ipv6-std-acl)# 5 deny any
```

To remove the ACL filter entry that will deny any IPv6 packets from the standard IPv6 access-list named my-list, enter the commands:

```
                awplus# configure terminal
        awplus(config)# ipv6 access-list standard my-list
awplus(config-ipv6-std-acl)# no deny any
```

Alternately, to remove the ACL filter entry with sequence number 5 to the standard IPv6 access-list named my-list, enter the commands:

```
                awplus# configure terminal
        awplus(config)# ipv6 access-list standard my-list
awplus(config-ipv6-std-acl)# no 5
```

Related Commands

ipv6 access-list standard (named)
show ipv6 access-list (IPv6 Software ACLs)
show running-config

show ipv6 access-list (IPv6 Software ACLs)

Use the **show ipv6 access-list standard** command to display a specified standard named IPv6 access-list that has been defined using the ipv6 access-list standard (named) command.

Syntax

```
show ipv6 access-list standard <access-list-name>
```

Parameter	Description
`standard`	Named standard access-list.
`<access-list-name>`	Specify an IPv6 access-list name.

Mode User Exec and Privileged Exec

Example To show the ipv6 access-list specified with the name `acl_name` use the following command:

```
awplus# show ipv6 access-list standard acl_name
```

Output Figure 34-1: Example output from the **show ipv6 access-list** command

```
Named Standard IPv6 access-list name
  deny any
```

Related Commands

ipv6 access-list extended (named)
(ipv6 access-list extended IP protocol filter)
(ipv6 access-list extended TCP UDP filter)
ipv6 access-list standard (named)
(ipv6 access-list standard filter)

Chapter 35: Quality of Service (QoS) Introduction

Introduction

This chapter introduces the concept of Quality of Service (QoS) with particular reference to Allied Telesis switches running the AlliedWare Plus^{TM} Operating System.

The concept of QoS is a departure from the original networking concept of treating all network traffic in the same way. Without QoS, all traffic types are equally likely to be dropped when a link becomes oversubscribed. With QoS, certain traffic types can be given preferential treatment. QoS is therefore a very useful tool both to control congestion and to meter or cap data in order to apply pre-agreed service levels.

Operationally, QoS is applied within the link and network layers. Functionally it provides the capability to intelligently transport your network traffic in order to provide stable and predictable end-to-end network performance.

Business benefits

Quality of Service mechanisms enable:

- network service providers to sell different levels of service to customers, based on what their customers require, and be confident in their ability to guarantee the reliable delivery of these services
- enterprise and educational organizations to actively manage and provide many services across one network, for example live video streaming and standard data services, with preferential treatment being given to mission-critical traffic
- network administrators to manage network congestion as network traffic levels increase and time-critical applications, such as streaming media, become more widely in demand by customers and organizations

QoS Operations

Quality of Service is typically based on how the switch performs the following functions:

- assigns priority to incoming frames (that do not already carry priority information)
- correlates prioritized frames with traffic classes, or maps frames to traffic classes based on other criteria
- correlates traffic classes with egress queues, or maps prioritized frames to egress queues
- provides minimum and maximum bandwidths for traffic classes, egress queues, and/or ports
- schedules frames in egress queues for transmission (for example, empty queues in strict priority or sample each queue)
- re-labels the priority of outgoing frames
- determines which frames to drop or re-queue if the network becomes congested
- reserves memory for switching/routing or QoS operation (for example, reserving buffers for egress queues or buffers to store packets with particular characteristics)

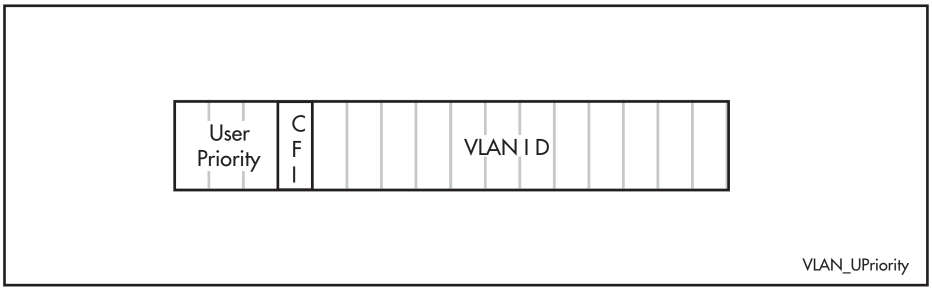
QoS Packet Information

Provision for QoS information to be embedded within the data fields exists within both the data link and network layer protocols. This information can then be used to assess the priority of the data and the resource preferences that need to be applied. The process of applying these service quality tags to your data is known as marking.

Link Layer QoS

Link layer frames entering a port may either be tagged or untagged. VLAN tagged frames contain the additional 802.1Q tag fields shown in Figure 35-1 below. Located within the TCI is a three bit User Priority field. This field is specifically provided to attach QoS based priority information, often referred to as the Class of Service (CoS) field.

Figure 35-1: IEEE 802.1Q Tagging



Appendix G of the IEEE Standard 802.1D provides some useful guidelines on applying priorities to 7 traffic types: These are summarized in the Table 35-1 below:

Table 35-1: CoS Traffic Mapping Guidelines

User Priority	Traffic Types
1	Background
2	Spare
0	Best Effort
3	Excellent Effort
4	Controlled Load
5	Video <100 ms latency and jitter
6	Voice <10 ms latency and jitter
7	Network Control

On the switch you can use the match cos command to select frames that match a particular User Priority value and assign them to a particular class-map. You can then map these incoming frames to one of eight egress queues. This facility enables you to accept frames that are already carrying meaningful priority information and automatically assign them to an appropriate egress queue. For example, you could decide to send frames with a User Priority value of 7 to queue 3, and frames with a User Priority value of 2 to queue 7. The process of assigning queues based on CoS tags is commonly known as “PreMarking”.

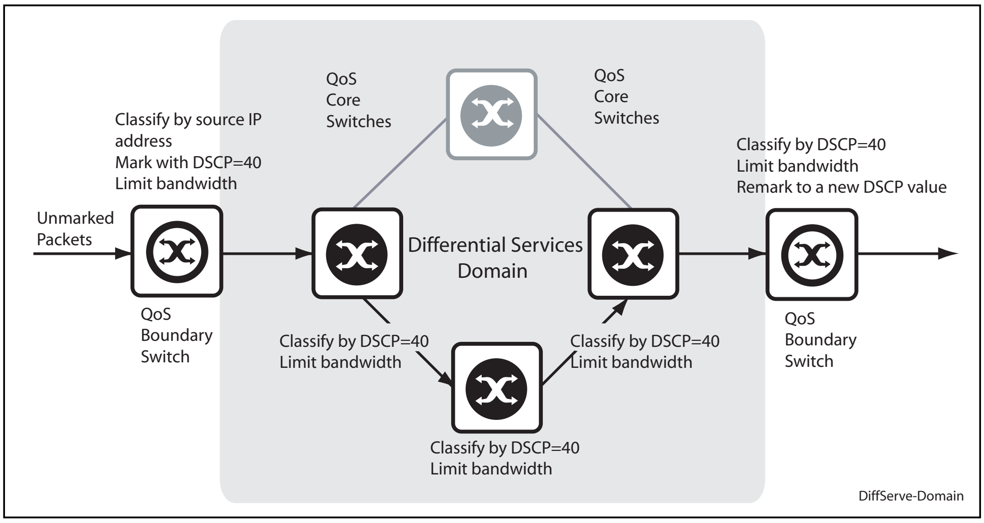
Note You configure the pre-marking steps to an ingress port. This process marks the data packets so that when they reach the egress port the decisions made during pre-marking can be applied in accordance with the configuration of the egress port.

Differentiated Services Architecture

Whilst a full description of the differential services model is outside the scope of this software reference, a brief introduction is provided. For further information, RFC 2475 provides an in depth definition of the architecture.

The basic differential services model envisages a multi router network within which common service qualities are applied. At the network boundary, *QoS Edge Routers* inspect the traffic and classify it into common service quality groups called Per Hop Behaviors (PHBs). A specific marker value called a Differential Services Code Point (DSCP) is added to the IP header of each packet, which allocates it to a PHB. *QoS Core Routers* within the network can then use the DSCP to decide on an appropriate service quality level to apply. When a network contains a consistently applied differential services code points DSCP it is referred as a Differential Services Domain (often shortened to DiffServe Domain). Figure 35-2 shows a simple Differential Services Domain.

Figure 35-2: Differentiated Services Domain

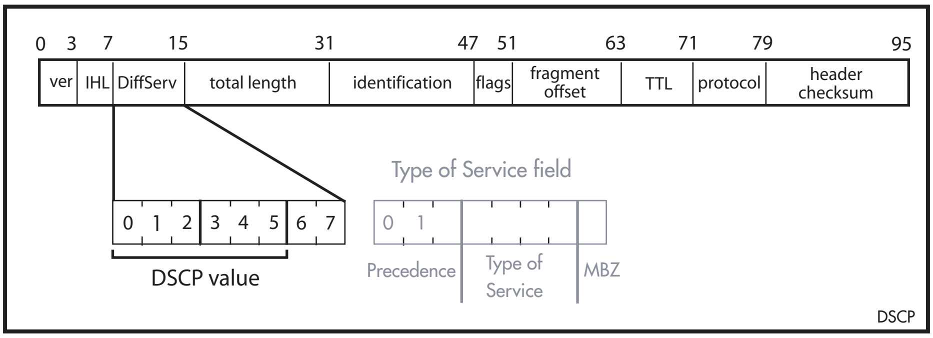


The Differential Services Field

Figure 35-3 shows an IP header containing a Differentiated Services field. The format of this redefined field is explained in RFC 2474; the main difference being that the old ToS field has been replaced by a 6 byte Differentiated Services Code Point (DSCP) field, which now provides for up to 64 defined values.

By applying this model only the QoS edge routers need to fully interrogate the incoming data packets; the QoS core routers are then relieved of this processing task and need only to inspect the DCSP before applying its appropriate forwarding, queueing, and shaping rules.

Figure 35-3: The DSCP bits of the DS field in the IPv4 header



On the switch you can use the match inner-vlan command to select frames containing a particular DSCP value, and associate them with a particular class map and policy map.

Because the model offers considerable flexibility, and the mapping of traffic types to DCSPs is individual for each network, this locally applied definition is known as a *Differential Services Domain*. The previous section introduced the concept of a Per Hop (service quality) Behaviors or PHBs. RFC 2597 defines a specific PHB group called Assured Forwarding (AF). The AF PHB group provides delivery of IP packets in four independently forwarded AF classes. Within each AF class, an IP packet can be assigned one of three different levels of drop precedence. Table 35-2 shows a list of recommended AF code points.

Table 35-2: Recommended DSCP Code Points

	(Lowest Priority) **Class 1 (001xxxx)**	**Class 2 (010xxxx)**	**Class 3 (011xxxx)**	(Highest Priority) **Class 4 (100xxxx)**
Low Drop Precedence	001010 AF11 Decimal 10	010010 AF21 Decimal 18	011010 AF31 Decimal 26	100010 AF41 Decimal 34
Medium Drop Precedence	001100 AF12 Decimal 12	010100 AF22 Decimal 20	011100 AF32 Decimal 28	100100 AF42 Decimal 36
High Drop Precedence	001110 AF13 Decimal 14	010110 AF23 Decimal 22	011110 AF33 Decimal 30	100110 AF43 Decimal 38

Processing pre-marked packets

A logical question to ask at this point is; how does the QoS switch deal with data that arrives with a pre-existing service level tag such as a DSCP? As previously touched on, the differentiated services model envisages a network that comprises QoS boundary routers at its edge and QoS core routers in its core network.

At the network edge the QoS boundary routers filter the incoming data based on specific packet components. Based on this filtering each packet is assigned a DSCP value. This value will determine the service level - priority, queueing etc - that will be applied.

Within the network core, the packet filtering required is reduced to simply reading the DSCP within each incoming packet, and applying the appropriate set of service levels. This relieves the core routers of the processing overhead of applying complex filtering to its high speed data streams.

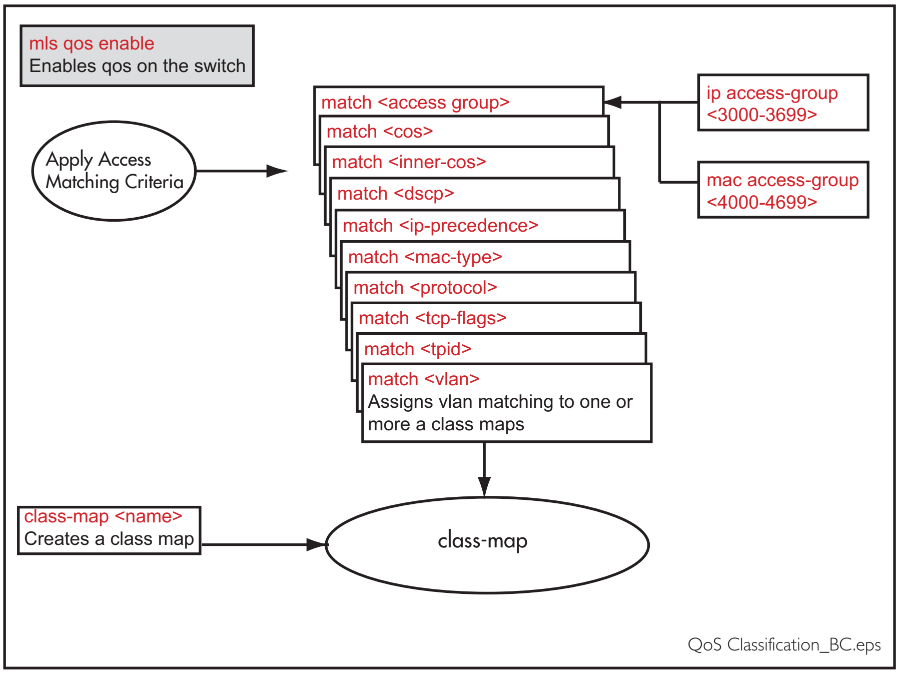
Applying QoS on Your Switch

This section steps you through the various stages of QoS set-up and introduces the QoS commands and how to apply them. Note that before you can configure any QoS functions on your switch, you must first enable QoS by using the mls qos enable command.

Classifying your Data

One of the early steps in setting up QoS on a network is planning and applying your classification rules. Classification is the process of **Filtering** and **Marking**. Filtering involves sorting your data into appropriate traffic types. Marking involves tagging the data so that downstream ports and routers can apply appropriate service policy rules. Figure 35-4 illustrates the classifying process, and will be referred to in the examples that follow.

Figure 35-4: QoS Classification Process



At the premarking stage you can assign your data a particular priority level by giving it a link level user priority, see "Link Layer QoS" on page 35.3, or a network level DSCP "Differentiated Services Architecture" on page 35.4. You can also assign the data to a particular output (or egress) queue.

Class Maps

Class Maps are among the pivotal QoS components. They provide the means that associate the classified traffic with its appropriate QoS actions. They are the linking elements for the following functions:

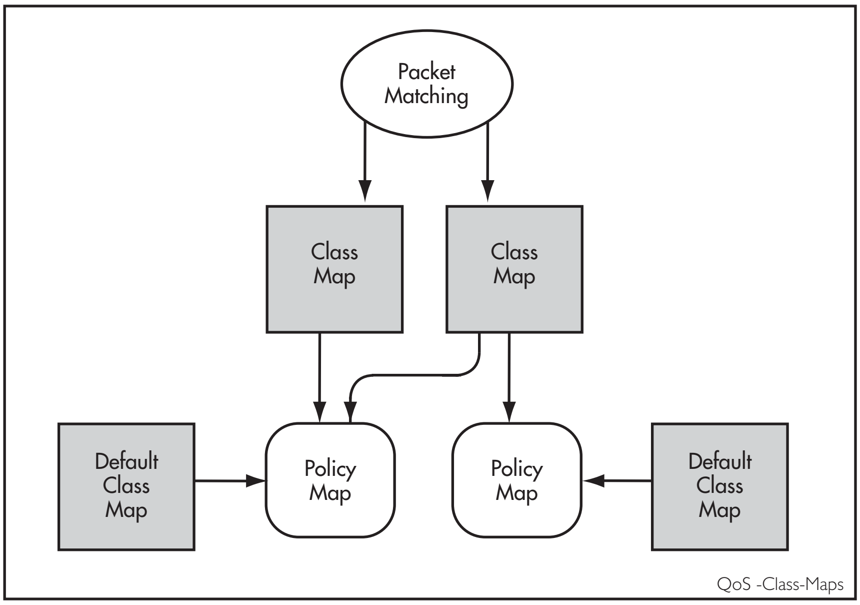
- classification
- policy mapping
- pre-marking

Figure 35-5 shows the relationship between a class-map and its associated functions. Note that the relationship between a class-map and a policy-map can be one-to-one or many-to-one. For information on policy-maps see the section, "Policy Maps" on page 35.10.

Note If a conflict occurs between the settings in two class maps, priority will be applied to the class map that was created first.

An example of such a conflict is the arrival of a packet that meets the classification requirements of two class maps each configured to the same policy map and set to apply different priority settings to the packet.

Figure 35-5: Relationship between a class map and its associated functions



Creating a class-map

To create a class map, use the class-map command on page 36.3.

This example creates a class-map called `video-traffic` and another called `data-traffic`:

```
           awplus#  configure terminal
   awplus(config)#  class-map video-traffic
awplus(config-cmap)#  exit
   awplus(config)#  class-map data-traffic
awplus(config-cmap)#
```

Creating and configuring default class-maps

These (automatically created) default class-maps serve as the means to specify the action that will apply to all unclassified data, i.e. all data within a policy-map that is not captured by any of the applied match commands that are applied to the policy-map by its class-maps.

Each time a new policy-map is created a new class map called "default" is also automatically created and assigned to the new policy map. You can configure any of the default class maps by using the default-action command on page 36.4

To set the default class-map for the policy-map p-map1 to have the action of deny:

```
         awplus# config
 awplus(config)# policy-map p-map1
awplus(config-pmap)# default-action deny
```

Applying a match command to a class-map

To apply a matching filter to a class map use one of the match commands.

This example creates a filter to select VLAN 5 traffic and applies this filter to the class map named video-traffic.

```
         awplus# config terminal
 awplus(config)# class-map video-traffic
awplus(config-cmap)# match vlan 5
```

Associating a class-map with a policy-map

To associate a class map with a policy map, use the class command on page 36.2.

Note A maximum of 128 class maps may be attached to each policy map.

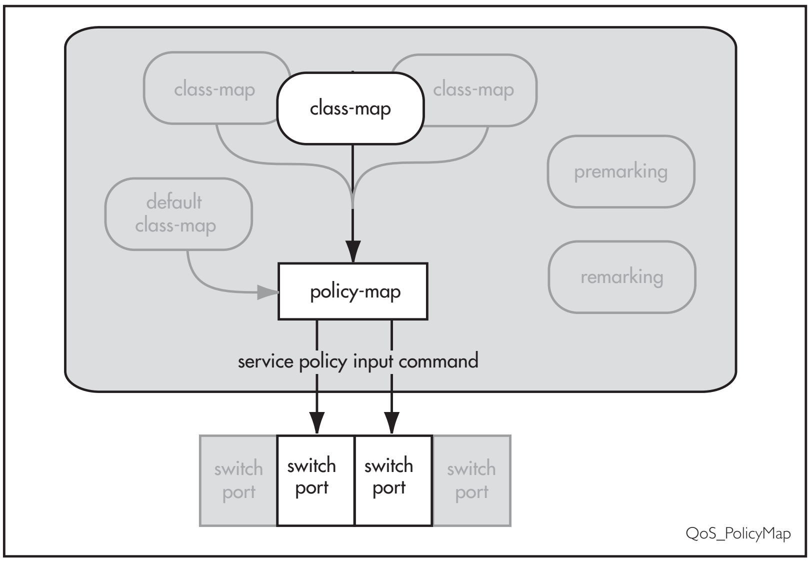
The following example creates a policy map called policy-one, and associates it with the class-maps named video-traffic, and database-traffic:

```
            awplus# configure terminal
    awplus(config)# policy-map policy-one
  awplus(config-pmap)# class video-traffic
awplus(config-pmap-c)# exit
  awplus(config-pmap)# class database-traffic
awplus(config-pmap-c)#
```

Policy Maps

Policy maps are the means by which you apply your class-map properties to physical switch ports. Figure 35-8 on page 35.16 illustrates this concept. Note that whilst a policy map can be assigned to several ports, a port cannot have more than one policy-map assigned to it.

Figure 35-6: Policy Maps and Related Entities



To create and name a new policy map you use the policy-map command on page 36.28.

To create a policy-map called pmap1 use the commands:

```
        awplus# configure terminal
awplus(config)# policy-map pmap1
```

Having created the policy map pmap1 we can use the class command on page 36.2 to assign it to one or more class maps. Since we created the class-maps video-traffic and office-traffic earlier in this chapter, we can now attach the policy-map pmap1 to both class-maps.

Use the class command to assign the policy map pmap1 to the class-maps video-traffic and office-traffic:

```
            awplus# configure terminal
    awplus(config)# policy-map pmap1
  awplus(config-pmap)# class video-traffic
awplus(config-pmap-c)# exit
  awplus(config-pmap)# class office-traffic
awplus(config-pmap-c)#
```

Premarking and Remarking Your Traffic

Premarking relates to adding QoS markers to your incoming data traffic before it is metered (policed). Remarking is the same process when applied after metering. Network switches will often be configured with two different premarking profiles, one for the QoS edge switches and another for the QoS core switches. This situation would apply if you are operating DSCP domains.

QoS markers can be applied at both the link layer (within the CoS field), and at the network layer (within the DSCP field). For more information on this topic see "QoS Packet Information" on page 35.3.

For boundary QoS switches

Traffic entering QoS boundary switches is unlikely to contain pre-existing QoS tagging. In this case, you can apply one or more of the following QoS mapping options.

- Assign a CoS tag to data associated with a particular class-map.
- Use the trust dscp command to enable the mls qos map premark DSCP map. This map enables you to change the DSCP tag and also map the tag to an egress port queue, a CoS value, or both. At the premarking stage you can set this mapping using the command, mls qos map premark-dscp to. After policing, you can then use the remark-map command to change the DCSP based on the packet's bandwidth class, or remap the existing bandwidth class, to a new value.

For an untagged packet, if no other mapping is applied and the packet is untagged, (i.e. in the absence of any other queue selection) traffic will be sent to queue 2.

For core QoS switches

Traffic entering ports within the QoS core network will almost certainly contain some pre-existing QoS tagging. Where this is the case, you can apply one of the following QoS mapping options.

- Map the CoS tag to an egress queue. You can do this either for the whole switch or for specific ports via their assigned policy-maps. See "CoS to egress queue premarking" on page 35.11.
- Map the DSCP tag to an output queue. You can do this either for the whole switch or for specific ports via their assigned policy-maps.
- Remap incoming data DSCP or CoS tags to values that are more appropriate for a particular switch or network.
- Assign bandwidth classes for your packets, based on the incoming DSCP. See Setting the Trust DSCP Map command on page 35.14.

CoS to egress queue premarking

If you are using CoS tagging for your QoS functions, your traffic is likely to be either entering the switch with a pre existing CoS tag, or will have appropriate tags attached via your class-maps and policy-maps. You can now mark the data for a particular egress queue, which will take effect when the data reaches its output port. There are two fundamental methods of applying CoS tagged packets to egress queues:

1. Apply a global mapping of CoS tags to egress queues for all ports.

2. Apply a CoS to egress queue mapping for the class-map / policy-map. This mapping - which forms part of the policy map - is applied at an input port, but will take effect at the packet's destination output port. Note that this procedure takes priority over that described in method (1) above.

These methods and their related commands will be now be described in greater detail.

CoS tagging commands

Table 35-3 shows the commands you can use to change the CoS field within incoming packets.

Table 35-3: CoS Mapping Commands in Hierarchical Order

Command	Function
mls qos map premark-dscp to	Where a packet contains CoS tag and a DSCP tag. The table set by this command contains a configurable DSCP to CoS tag mapping.
remark-map	Configures the remark map. This command is applied when a policer is configured with the **action** parameter of the command, police twin-rate action set to **remark-transmit.**

Note Where a packet contains both a CoS and a DSCP field, and each field maps to a different class-map; the switch will apply a priority that is based on the date that the class map was added to the policy map; the earlier the date, the higher the priority.

Mapping CoS tags to traffic types

The command mls qos map cos-queue to, enables you to create a switch-wide mapping of CoS values to egress queues. The default mappings for this command are:

```
COS :           0 1 2 3 4 5 6 7

--------------------------------

QUEUE:          2 0 1 3 4 5 6 7
```

These mappings match the CoS guidelines documented in Annex H.2 of ANSI/IEEE 802.1D 1988 Edition. Table H-15 on page 355 of the standard, shows a table of user priorities for specific traffic types. Table 35-4 shows an adapted version of the ANSI/IEEE table.

Table 35-4: Traffic Type Guidelines

User Priority (egress queue)	CoS Value	Acronym	Traffic type	Internal Traffic Queue Defaults
0 (lowest)	1	BK	Background	
1	2	-	Spare	
2	0	BE	Best Effort	Default
3	3	EE	Excellent Effort	
4	4	CL	Controlled Load	
5	5	VI	"Video," <100 ms latency and jitter	
6	6	VO	"Voice," <10 ms latency and jitter	EPSR-Management BPDU ARP-Requests
7 (highest)	7	NC	Network Control	Stack Management

CoS settings for VCStack stack operation

In general you can apply the same principles when configuring QoS on a VCStack as you would for single switch; however there are a few specific changes that you will need to make.

Switches within a VCStack, exchange their stack management information and user data over their high speed inter-stacking links. The stack management information is pre-assigned to the egress queue 7. This is the highest value queue, and (in a stacked configuration) its traffic should not be shared with any user data. However, any CoS tagging of 7 applied to the incoming data will automatically be assign to queue 7 as it crosses the internal stacking links. You will therefore need to reconfigure your CoS to Queue settings to ensure that no user data is sent to queue 7.

To prevent this from happening, we recommend that you make appropriate changes to your queue settings (mappings) to reflect the stacking requirement previously described. For more information on this topic, see "Mapping CoS tags to traffic types" on page 35.12.

This process should include (but not be limited to) running the following command to ensure that any remaining user still carrying a CoS 7 tag, will be mapped to egress queue 6.

To remap priority CoS traffic to egress queue 6, run the following command.

```
        awplus#  config terminal
awplus(config)#  mls qos map cos-queue 7 to 6
```

DSCP to egress queue premarking

If you are using DSCP tagging for your QoS functions, your traffic is likely to be entering the switch either with a pre existing DSCP tag, or will have appropriate DSCP tags attached via your class-maps and policy-maps. You can now mark the data for a particular egress queue, which will take effect when the data reaches its output port.

If your switch forms part of a DSCP domain, you can adapt the steps in this section to apply the mappings and settings to match the standards you have selected for the domain. This mapping - which forms part of the policy map - is applied at an input port, but will take effect at the packet's destination output port.

DSCP to egress queue premarking commands

A number of commands can be used for mapping DSCP tags. Where these conflict, the switch applies a pre-defined set of priorities. Table 35-5 lists these priorities in order (lowest priority first).

Where a packet that contains both CoS and a DSCP fields and each field maps to a different class-map / policy-map, the switch will apply a priority based on the creation date of class maps - the earlier the creation date, the higher the priority priorities.

Table 35-5: DSCP Mapping Commands in Hierarchical Order

Command	Function
trust dscp	Setting the trust dscp enables the mls qos map premark-dscp to command to apply. See, "Setting the Trust DSCP Map" on page 35.14.
mls qos map premark-dscp to	With the trust dscp set, this command applies a remapping table whose values include the dscp and egress queues.

Setting the Trust DSCP Map

The Trust DSCP mapping table assigns a new set of QoS values for a DSCP value supplied as table input. To configure this table you use the mls qos map premark-dscp to command.

Table 35-6: Drop Probability Table

Table Input	-- Table Output			------------------------------------
Existing DSCP	New DSCP Value	New CoS Value	New Queue No	New BW Class green yellow red

The Trust DSCP map provides the highest priority of all the pre-marking controls. To apply this table you must first apply the trust setting by using the trust dscp command.

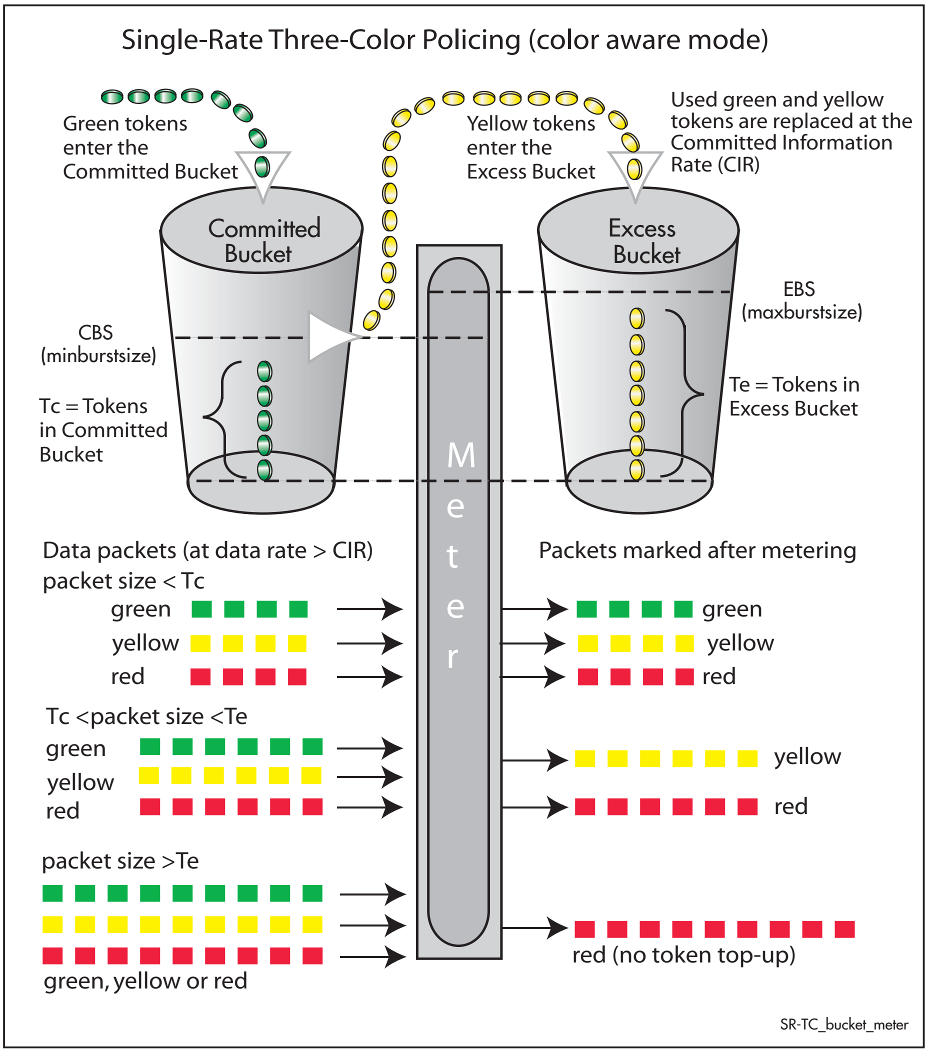
Policing (Metering) Your Data

Once you have set-up your classification and created your class-maps, you can start conditioning your traffic flows. One tool used for traffic conditioning is the policer (or meter). The principle of policing is to measure the data flow that matches the definitions for a particular class-map; then, by selecting appropriate data rates, allocate the flows into one of three categories: Red, Yellow, or Green. You then decide what action to apply to the Red, Yellow and Green data.

Single-rate Three-color Policing

This policing method is based on that defined in RFC 2697. The principle of single-rate three-color policing is shown in Figure 35-7. For a given class-map, a meter monitors both the token count in the buckets, and the input data flow.

Figure 35-7: Single-rate Three-color Policing


Single-Rate Three-Color Policing (color aware mode)
Green tokens enter the Committed Bucket
Yellow tokens enter the Excess Bucket
Used green and yellow tokens are replaced at the Committed Information Rate (CIR)
Committed Bucket
Excess Bucket
CBS (minburstsize)
EBS (maxburstsize)
Tc = Tokens in Committed Bucket
Te = Tokens in Excess Bucket
Meter
Data packets (at data rate > CIR)
packet size < Tc
green
yellow
red
Packets marked after metering
green
yellow
red
Tc <packet size <Te
green
yellow
red
yellow
red
packet size >Te
green, yellow or red
red (no token top-up)
SR-TC_bucket_meter

Each byte entering the meter is paired with a token in one of the buckets, and a token is removed as each byte is accepted. If the input data rate is the same as the CIR then the data passes through the port at the same rate as its bucket fills. Hence the bucket level remains constant. In this model, the data buffer is represented by two data buckets. You can specify the CIR using the police single-rate action command.

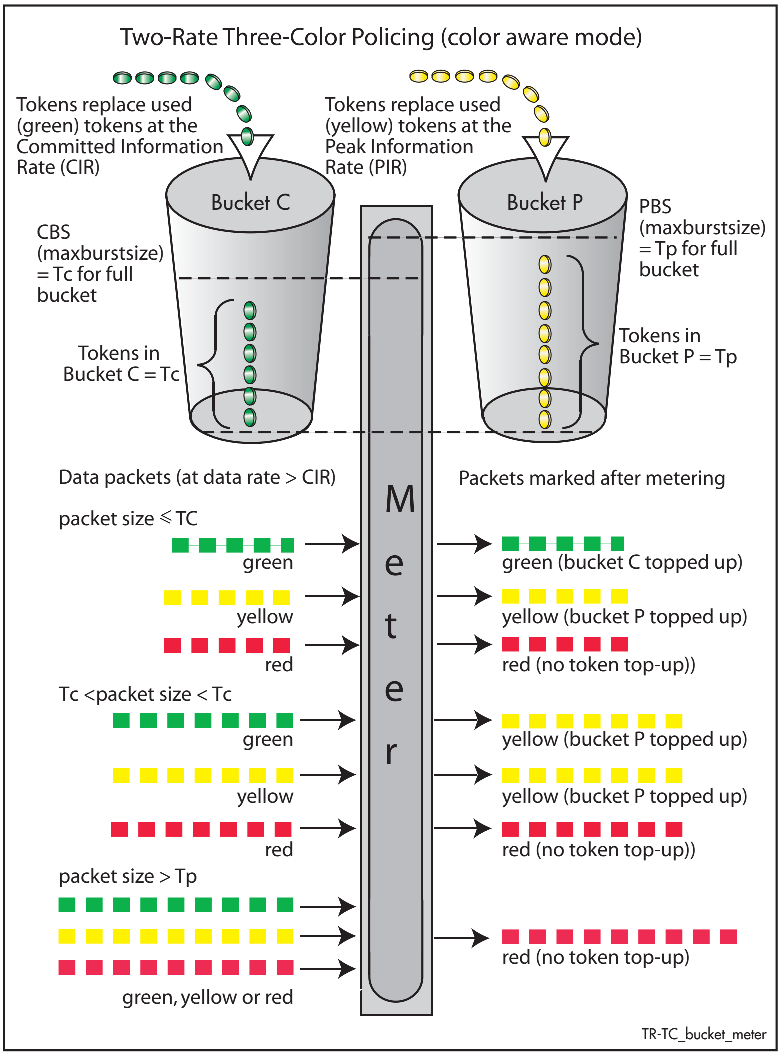
Initially both buckets have their full token count. A surge of date exceeding the CIR will begin to empty the bucket. As the data and tokens are paired, data bytes that match tokens below the CBS level are marked green, those that are between CBS and EBS will be marked yellow, and those that are above EBS are marked red.

Note that although the data is metered per byte, the color marking process is applied per packet. This means that if there were only sufficient tokens available to match part of a packet, then the whole packet would be marked red. Then, depending on the **action** parameter of the police single-rate action command, the whole packet will be either dropped or forwarded. In either situation, the red marked packet will leave the bucket counts unchanged.

Two-rate Three-color Policing

This policing method is based on that defined in RFC 2698. The principle of two-rate three-color policing is shown in Figure 35-8.

Figure 35-8: Two-rate Three-color Policer


Two-Rate Three-Color Policing (color aware mode)
Tokens replace used (green) tokens at the Committed Information Rate (CIR)
Tokens replace used (yellow) tokens at the Peak Information Rate (PIR)
Bucket C
Bucket P
CBS (maxburstsize) = Tc for full bucket
PBS (maxburstsize) = Tp for full bucket
Tokens in Bucket C = Tc
Tokens in Bucket P = Tp
Meter
Data packets (at data rate > CIR)
Packets marked after metering
packet size ⩽ TC
green
yellow
red
green (bucket C topped up)
yellow (bucket P topped up)
red (no token top-up))
Tc <packet size < Tc
green
yellow
red
yellow (bucket P topped up)
yellow (bucket P topped up)
red (no token top-up))
packet size > Tp
green, yellow or red
red (no token top-up)
TR-TC_bucket_meter

For a given class-map, the meter monitors the token count in both buckets, and the input data flow. Initially tokens enter both buckets until full. As the data enters a port, the meter pairs each byte to a token in one of the buckets, then removes a token from the appropriate bucket. Bucket C is topped up with tokens at the Committed Information Rate (CIR), and bucket P is topped up at the Peak Information Rate (PIR).

When data enters the port at the CIR, the bucket fills at the same rate as the incoming data, thus the token count in bucket C remains constant. Similarly, if data enters the port at the PIR, then the token count in bucket P remains constant. You can specify the CIR and the PIR by using the police twin-rate action command. The function of each of this command is explained in the section "Configuring and Applying a Policer" on page 35.17.

A surge of data exceeding the CIR will begin to empty bucket C. If bucket C empties to a point where it has insufficient tokens to match to an incoming data packet, then the data packet will be marked yellow. The data will now be measured against the level in bucket P and tokens will be removed from this bucket to match the incoming data. If the incoming data rate drops to less than the CIR then the data will continue to be marked yellow until the level in bucket C has had a chance to fill, whereupon it will be marked *green*.

If the incoming data is greater than the PIR, then bucket P begins to empty. If bucket P empties to a point where it has insufficient tokens to match to an incoming data packet, then the data packet will be marked *red*. In this situation no tokens are removed from either bucket.

Note that although the data is metered per byte, the color marking process is applied per packet. This means that if there were only sufficient tokens available to match part of a packet, then the whole packet would be marked red. Then, depending on the **action** parameter of the police twin-rate action command, the whole packet will be either dropped, or marked and forwarded. In either situation, the red marked packet will leave the bucket counts unchanged.

Configuring and Applying a Policer

The previous section showed how the policer works and how to select either the single rate or twin rate action. To apply a policy to class maps:

- Select your policy-map and class-map from the command prompt, then enter either the police single-rate action command or the police twin-rate action command whilst selecting the appropriate command parameters.

This will apply the command to the selected class-map. By running this command several times, each for a different class-map, you can apply separate meter settings to each class-map.

Remarking Your Data

The remarking process enables you to change the QoS tagging and queue assignments etc from data that has already been marked by the policer. To do this you fill entries in the remarking table by using the remark-map command on page 36.30. In order to remark your data ensure that the **action** parameter of either the police single-rate action or the police twin-rate action is set to **remark-transmit**.

The following table shows the remarking options

Table 35-7: Remarking Table

BANDWIDTH CLASS		
Green	New DSCP	New bandwidth class (Red, Yellow, or Green)
Yellow	New DSCP	New bandwidth class (Red, Yellow, or Green)
Red	New DSCP	New bandwidth class (Red, Yellow, or Green)

Example Traffic presently marked either Yellow or Red is to be remarked green and assigned a new DSCP value of 25:

Table 35-8: Remarking Table Example

BANDWIDTH CLASS		
Yellow	New DSCP = 25	New bandwidth class =Green

To configure this setting, you would enter the following commands:

```
            awplus# configure terminal
    awplus(config)# policy-map pmap1
awplus(config-pmap)# class cmap1
awplus(config-pmap-c)# remark-map bandwidth-class green to
                       new-dscp 25 new-bandwidth-class yellow
```

Further remarking can be achieved by using the remark new-cos command on page 36.32. This command enables you to configure and remark either or both the CoS flag in the data packet, and the input into the CoS to queue map thus changing the destination egress queue.

Configuring the Egress Queues

Previous sections have explained the ingress functions. These include, how the incoming data can be classified and marked according to its priority and allocated to an egress queue, then finally how metering and remarking is applied. At this point the data then flows across the switch to its destination egress port where its transit to the egress queues is controlled.

The means by which data is applied to the egress queues is dependant on three functions:

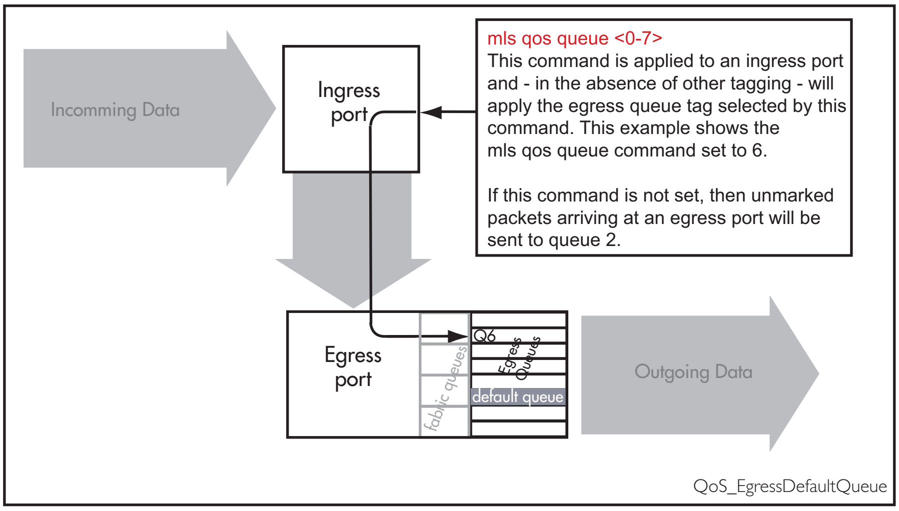
- Egress queue and QoS markers that are set within each data packet
- Egress controls that are applied to the whole switch
- Egress that are applied to each individual switch port

Egress Queues and QoS markers

Once the data packets have been appropriately filtered, classified, policed, and remarked, they travel across the switch's internal paths carrying their assigned QoS tag markers such as their priority, class and destination queues. For more details on ingress data marking, refer to the earlier sections of this chapter. At the egress port these markers are read and used to determine which queues each data packet will be forwarded to, and the priorities that will be applied.

There are eight egress queues allocated to each egress port. The egress queue that a particular packet passes through is determined by either the configuration of the switch, or the markers contained within the packet itself.

Figure 35-9: Default Egress Queue



mls qos queue <0-7>
This command is applied to an ingress port and - in the absence of other tagging - will apply the egress queue tag selected by this command. This example shows the mls qos queue command set to 6.

If this command is not set, then unmarked packets arriving at an egress port will be sent to queue 2.

Egress Queue Commands Hierarchy

The destination queue that any one packet will take depends on the markers within the packet, and the way the queueing commands have been set. Also, some queueing commands will override others. Here is how the switch prioritizes its queueing commands.

Imagine a packet entering an ingress port then traveling through the switch fabric to reach its appropriate egress port. In this situation the following hierarchy will apply:

1. If the packet enters an egress port carrying no QoS markers and no QoS queueing commands have been set on the switch, then the packet will exit the port via queue number 2.

2. If the packet containing CoS marker arrives at an egress port, then with no other configuration applying, then its queue mapping will be subject to the setting of the mls qos map cos-queue to command.

3. Situations (1) and (2) can be overridden by the remark new-cos command. This command sets a default queue for each switch port.

Egress Queue Shaping

This section is concerned with how the egress queues are cleared.

Strict priority servicing

By default, all queues on all ports are serviced in a strict priority order. This means that the highest numbered priority queue (queue 7) is emptied first; then when it is completely empty, the next highest priority queue is processed, and so on. Thus, for a strict priority queue to be processed, all higher priority queues must be empty.

Strict priority servicing is the default setting; however if your system is configured for weighted round robin (WRR), you can return it to priority queueing by using the commands shown in the following example.

To return queue 3 of `port1.0.1` from WRR servicing to strict priority queueing, use the commands:

```
        awplus# configure terminal
awplus(config)# interface port1.0.1
awplus(config-if)# priority-queue 3
```

Weighted round robin servicing

The following examples show how to configure round robin servicing.

Example To configure a wrr-queue by applying a weighting value of 6 to queues 0 1 2:

```
        awplus# configure terminal
awplus(config)# interface port1.0.1
awplus(config-if)# wrr-queue weight 6 queues 0 1 2
```

Example In this example port 1.0.1 has queues configured as follows:

- queues 6 and 7 are configured strict priority
- queues 3 4 and 5 are configured as WRR with weighting values of 6,
- queue 5 is configured as WRR with weighting values of 12
- queues 0, 1 and 2 are configured as WRR with weighting values of 4,

```
        awplus# configure terminal
awplus(config)# interface port1.0.1
awplus(config-if)# priority-queue 6 7
awplus(config-if)# wrr-queue weight 6 queues 3 4
awplus(config-if)# wrr-queue weight 12 queues 5
```

In this example, the queues are processed as follows:

1. Queue 7 is processed first.

2. If queue 7 is empty, Queue 6 is processed next.

3. If queues 6 and 7 are empty, queue 5 is processed next.

4. If queue 5 is empty, queues color 3 and 4 are processed with equal weighting.

Drop Mode

The drop mode sets the limits for packets in the eight egress queues and determines how packets will be dropped if the queues become congested. Your switch supports the Tail Drop mode and is pre-configured with the following defaults:

Data packets will be dropped per color at the following buffer usage:

Red at 60%, Yellow at 80%, and Green at 100%.

These settings cannot be reconfigured.

Tail Drop

In this drop mode each egress queue is configured with a maximum threshold value. This value represents the point where the egress buffer queues are full and the egress port must start dropping data. The port does this by dropping data packets destined for the full queue on a "last in first dropped" basis. This enables the port to clear its data already queued for egress.

If a "reliable" transport protocol, such as TCP is used, this data should be retransmitted, but at a slower rate due to lack of returning "acknowledgements".

Storm Protection

Storm protection uses QoS mechanisms to classify on traffic likely to cause a packet storm (broadcast and multicast). Unless you are running an enhanced storm protection feature such as Loop Protection, the per-port storm protection mechanism simply discards any traffic over the configured limit. However, with QoS storm protection, several actions are possible when a storm is detected:

- You can disable the port physically.
- You can disable the port logically.
- You can disable the port for a particular VLAN.

Storm protection is activated as soon as a port is enabled, before the port forwards frames.

When a storm is detected on a port, a message is automatically recorded in the log, and you can configure an SNMP trap to signal that a port has been disabled. When a storm is detected on a trunk or port group, the entire trunk or port group is disabled.

The following table explains the basic concepts involved with storm protection.

Concept	Description
Window	The frequency at which traffic is measured to determine whether storm protection should be activated.
Rate	The amount of traffic per second that must be exceeded before the switch takes the configured action.
Action	What the switch does when it detects a storm on a port.
Timeout	The length of time the port remains disabled after a port has been disabled due to a packet storm.

To set the action to take when triggered by QoS Storm Protection (QSP), use the storm-action command on page 36.45.

To set the time to re-enable the port once disabled by QSP, use the storm-downtime command on page 36.46.

To enable the policy-based storm protection QSP, use the storm-protection command on page 36.47.

Policy-Based Routing

Policy based routing provides a means to create multiple paths to the same destination. The specific path that any particular packet will take can be based on configurable network metrics such as priority, protocol, or VLAN membership. For example, policy based routing can implement policies to allow or deny paths based on the identity of user devices, application, or packet sizes.

Practical Example

The example shown makes use of policy based routing to achieve the following:

1. Ensure that traffic being sent between local VLANs is switched normally.

2. Selects a particular egress path for traffic destined for the wide are networks.

Configuration Overview

A large government building houses employees from three separate government departments: Heath, Welfare, and Employment. Each department has its own local subnet, and an associated VLAN; these are:

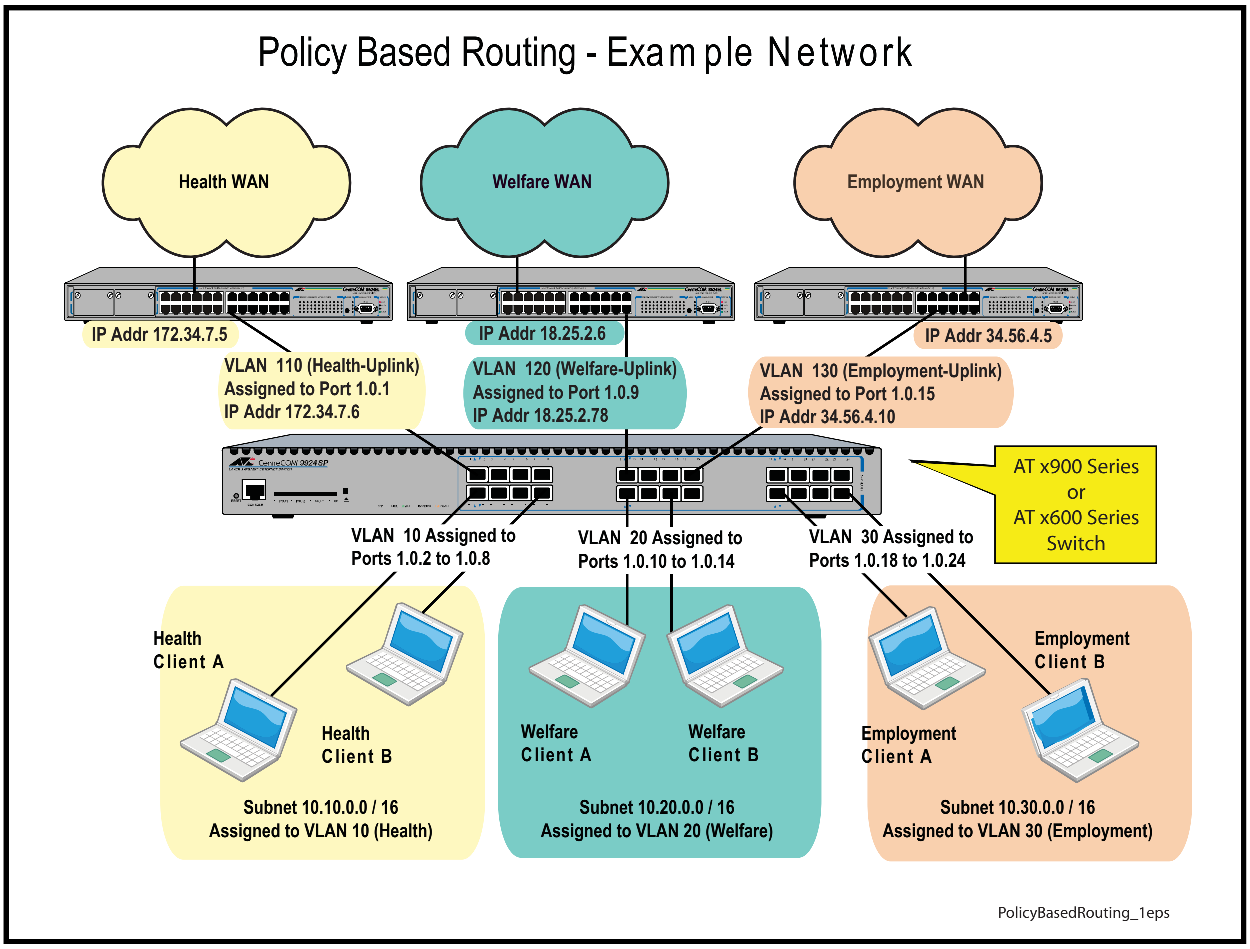
- 10.10.0.0/16 Health, VLAN 10
- 10.20.0.0/16 Welfare, VLAN 20
- 10.30.0.0/16 Employment, VLAN 30

Enquiries to each department are fed through a common Allied Telesis switch. The switch has 3 uplink ports, each of which (for simplicity) will be in a different VLAN and each will supply a connection to its relevant government department and to the Internet via each departments particular ISP (Internet Service Provider). These are:

- Port 1.0.1 Health Uplink, VLAN 110
- Port 1.0.5 Welfare Uplink, VLAN 120
- Port 1.0.21 Employment Uplink, VLAN 130

This configuration is illustrated in Figure 35-10:

Figure 35-10: Policy-Based Routing Example



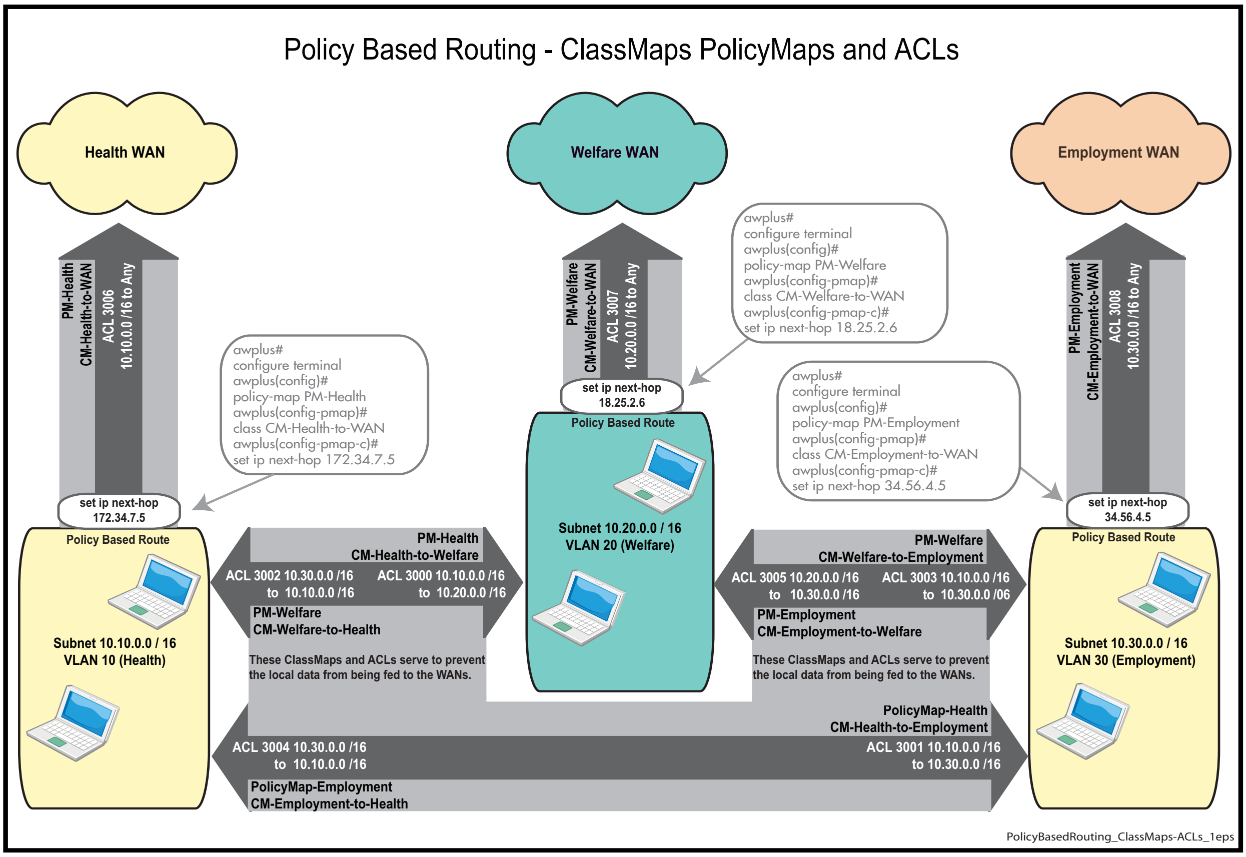
Configuration Steps

The following steps can be used to setup this example network. Since each step involves entering several instances of a command type, a single practical command entry is shown at the end of each step. The set of steps comprise the following:

1. Create VLANs on the switch.
2. Create access control lists (ACLs) that will match the data flows between local subnets.
3. Create ACLs that will match the data flows between local devices and other destinations.
4. Setup class-maps for each department and apply an access-list to each of the class maps.
5. Setup class-maps for each department's wide area connection and apply an access-list to each of these class maps.
6. Create the departmental policy-maps and associate them with their appropriate class-maps.
7. Apply these policy-maps to their appropriate ports.

These class-maps and ACLs are shown diagrammatically in Figure 35-11 below.

Figure 35-11: Policy Based Routing Example - ClassMaps and ACLs



```
awplus#
configure terminal
awplus(config)#
policy-map PM-Health
awplus(config-pmap)#
class CM-Health-to-WAN
awplus(config-pmap-c)#
set ip next-hop 172.34.7.5
```

```
awplus#
configure terminal
awplus(config)#
policy-map PM-Welfare
awplus(config-pmap)#
class CM-Welfare-to-WAN
awplus(config-pmap-c)#
set ip next-hop 18.25.2.6
```

```
awplus#
configure terminal
awplus(config)#
policy-map PM-Employment
awplus(config-pmap)#
class CM-Employment-to-WAN
awplus(config-pmap-c)#
set ip next-hop 34.56.4.5
```

Step 1: Create VLANs on the switch

- Create VLANs 10, 20, and 30
- Apply these VLANs to their appropriate local ports

Practical example: Create VLAN 10 and apply it to port1.0.2-port1.0.8.

```
        awplus# configure terminal
awplus(config)# interface port port1.0.2-port1.0.8
awplus(config)# switchport mode access
awplus(config)# switchport access vlan 10
```

- Create VLANs 110, 120, and 130
- Apply these VLANs to their appropriate WAN ports

Practical example: Create VLAN 110 and apply it to port1.0.1.

```
        awplus# configure terminal
awplus(config)# interface port port1.0.1
awplus(config)# switchport mode access
awplus(config)# switchport access vlan 110
```

Step 2: Create access control lists (ACLs) that will match the data flows between local user devices.

- access-list 3000 permit ip 10.10.0.0/16 10.20.0.0/16
  Matches for packets from the Health user devices to Welfare user devices.
- access-list 3001 permit ip 10.10.0.0/16 10.30.0.0/16
  Matches packets from the Health user devices to Employment user devices.
- access-list 3002 permit ip 10.20.0.0/16 10.10.0.0/16
  Matches packets from the Welfare user devices to Health user devices.
- access-list 3003 permit ip 10.20.0.0/16 10.30.0.0/16
  Matches packets from the Welfare user devices to Employment user devices.
- access-list 3004 permit ip 10.30.0.0/16 10.10.0.0/16
  Matches packets from the Employment user devices to Health user devices.
- access-list 3005 permit ip 10.30.0.0/16 10.20.0.0/16
  Matches packets from the Employment user devices to Welfare user devices.

Practical example: Create an ACL that matches packets from the Health user devices to Welfare user devices.

```
        awplus#  configure terminal
awplus(config)#  access-list 3000 permit ip 10.10.0.0/16
                 10.20.0.0/16
```

Step 3: Create access control lists (ACLs) that will match the data flows between user devices and all other destinations.

- access-list 3006 permit ip 10.10.0.0/16 any
  Matches packets from Health user devices to all other destinations.
- access-list 3007 permit ip 10.20.0.0/16 any
  Matches packets from Welfare user devices all other destinations.
- access-list 3008 permit ip 10.30.0.0/16 any
  Matches packets from Employment user devices to all other destinations.

Practical example: Matches packets from the Health user devices to all other destinations.

```
        awplus#  configure terminal
awplus(config)#  access-list 3006 permit ip 10.10.0.0/16 any
```

Step 4: Setup class-maps for each department and apply an access-list to each of the class maps.

- class-map CM-Health-to-Welfare
  Creates a class map called **CM-Health-to-Welfare**
- match access-group 3000
  Applies the access-list 3000 to the **CM-Health-to-Welfare** class-map, so that this class-map applies to all packets matching this ACL.
- class-map CM-Health-to-Employment
  Creates a class map called **CM-Health-to-Employment.**
- match access-group 3001
  Applies the access-list 3001 to the **CM-Health-to-Employment** class-map, so that this class-map applies to all packets matching this ACL.
- class-map CM-Welfare-to-Health
  Creates a class map called **CM-Welfare-to-Health.**
- match access-group 3002
  Applies the access-list 3002 to the **CM-Welfare-to-Health** class-map, so that this class-map applies to all packets matching this ACL.
- class-map CM-Welfare-to-Employment
  Creates a class map called **CM-Welfare-to-Employment.**
- match access-group 3003
  Applies the access-list 3003 to the **CM-Welfare-to-Employment** class-map, so that this class-map applies to all packets matching this ACL.
- class-map CM-Employment-to-Health
  Creates a class map called Employment-to-Health.
- match access-group 3004
  Applies the access-list 3004 to the **CM-Employment-to-Health** class-map, so that this class-map applies to all packets matching this ACL.
- class-map CM-Employment-to-Health
  Creates a class map called **CM-Employment-to-Health.**
- match access-group 3005
  Applies the access-list 3005 to the class-map **CM-Employment-to-Health** class-map, so that this class-map applies to all packets matching this ACL.

Practical example: Create the class-map CM-Health-to-Welfare, then apply access list 3000 to it.

```
          awplus# configure terminal
  awplus(config)# class-map CM-Health-to-Welfare
awplus(config-cmap)# match access-group 3000
```

Step 5: Setup class-maps for each department’s wide area connection and apply an access-list to each of these class maps.

- class-map CM-Health-to-WAN
  Creates a class map called CM-Health-to-WAN.
- match access-group 3006
  Applies the access-list 3006 to the **CM-Health-to-WAN** class-map, so that this class-map applies to all packets matching this ACL.
- class-map CM-Welfare-to-WAN
  Creates a class map called Welfare-to-WAN
- match access-group 3007
  Applies the access-list 3007 to the **CM-Welfare-to-WAN** class-map, so that this class-map applies to all packets matching this ACL.
- class-map CM-Employment-to-WAN
  Creates a class map called Employment-to-WAN.
- match access-group 3008
  Applies the access-list 3008 to the **CM-Employment-to-WAN** class-map, so that this class-map applies to all packets matching this ACL.

Practical example: Create the class-map CM-Health-to-Welfare, then apply access list 3006 to it.

```
            awplus# configure terminal
    awplus(config)# class-map CM-Health-to-WAN
awplus(config-cmap)# match access-group 3006
```

Step 6: Create the Departmental Policy-Maps and associate them with their appropriate Class-Maps.

- policy-map PM-Health
  Creates the policy-map called PM-Health
- class CM-Health-to-Welfare
- class CM-Health-to-Employment
  Attaches the local Health class-maps to the PM-Health policy-map. Note that no action is applied to these two class maps. Packets that match either of these two class-maps will be forwarded across the local network using normal routing / forwarding procedures.
- class CM-Health-to-WAN
- set ip next-hop 172.34.7.5
  Attaches the CM-Health-to-WAN class-map to this policy-map, and gives it a policy-routing action.
- policy-map PM-Welfare
  Creates the policy-map called PM-Welfare
- class CM-Welfare-to-Health
- class CM-Welfare-to-Employment
  Attaches the local Welfare class-maps to the PM-Welfare policy-map. Note that no action is applied to these two class maps. Packets that match either of these two class-maps will be forwarded across the local network using normal routing / forwarding procedures.
- class CM-Welfare-to-WAN

- set ip next-hop 18.25.2.6
  Attaches the Welfare-to-WAN class-map to this policy-map, and gives it a policy-routing action.
- policy-map PM-Employment
  Creates the policy-map called PM- Employment
- class CM-Employment-to-Health
- class CM-Employment-to-Welfare
  Attaches the local Employment class-maps to the PM-Employment policy-map. Note that no action is applied to these two class maps. Packets that match either of these two class-maps will be forwarded across the local network using normal routing / forwarding procedures.
- class CM-Employment-to-WAN
- set ip next-hop 34.56.4.5
  Attaches the Employment-to-WAN class-map to this policy-map, and gives it a policy-routing action.

Practical example: Create the policy-map called PM-Employment and attach its appropriate classmaps.

```
              awplus# configure terminal
      awplus(config)# policy-map PM-Employment
 awplus(config-pmap)# class CM-Employment-to-Health
awplus(config-pmap-c)# exit
 awplus(config-pmap)# class CM-Employment-to-Welfare
awplus(config-pmap-c)# exit
 awplus(config-pmap)# class CM-Employment-to-WAN
awplus(config-pmap-c)# set ip next-hop 34.56.4.5
```

Step 7: Apply these Policy Maps to appropriate ports.

- service-policy input Health-to-WAN
- service-policy input Welfare-to-WAN
- service-policy input Employment-to-WAN

Practical example: To apply a policy map named PM-Employment to `port 1.0.18-port1.0.24`:

```
         awplus# configure terminal
 awplus(config)# interface port1.0.18-port1.0.24
awplus(config-if)# service-policy input PM-Employment
```

Chapter 36: QoS Commands

Command List

This chapter provides an alphabetical reference for Quality of Service commands. For more information, see Chapter 35, Quality of Service (QoS) Introduction and Chapter 31, Access Control Lists Introduction.

class

Use this command to associate an existing class map to a policy or policy map (traffic classification), and to enter Policy Map Class Configuration mode to configure the class map.

Use the **no** variant of this command to delete an existing class-map.

For more information on class-maps and policy maps, see the following sections:“Class Maps” on page 35.7 and “Policy Maps” on page 35.10.

Note that if your class map does not exist, you can create it by using the class-map command.

Syntax

```
class {<name>|default}

no class <name>
```

Parameter	Description
<name>	Name of the (already existing) class map.
default	Specify the default class map.

Mode Policy Map Class Configuration

Example The following example creates the policy map pmap1(using the policy-map command), then associates this to an already existing class map named cmap1, use the commands:

```
           awplus# configure terminal
   awplus(config)# policy-map pmap1
awplus(config-pmap)# class cmap1
awplus(config-pmap-c)#
```

Related Commands class-map
policy-map

class-map

Use this command to create a class map.

Use the **no** variant of this command to delete the named class map.

Syntax

```
class-map <name>

no class-map <name>
```

Parameter	Description
<name>	Name of the class map to be created.

Mode Global Configuration

Example This example creates a class-map called `cmap1`, use the commands:

```
              awplus#  configure terminal
      awplus(config)#  class-map cmap1
 awplus(config-cmap)#
```

clear mls qos interface policer-counters

Resets an interface's policer counters to zero. This can either be for a specific class-map or for all class-maps.

Syntax

```
clear mls qos interface <port> policer-counters
    [class-map <class-map>]
```

Parameter	Description
<port>	The port may be a switch port (e.g. `port1.0.4`), a static channel group (e.g. `sa3`), or a dynamic (LACP) channel group (e.g. `po4`).
`class-map`	Select a class-map.
<class-map>	Class-map name.

Mode Privileged Exec

Example To reset the policy counters to zero for all class maps for `port1.0.1`, use the command:

```
awplus# clear mls qos interface port1.0.1 policer-counters
```

Related Commands show mls qos interface policer-counters

default-action

Sets the action for the default class-map belonging to a particular policy-map. The action for a non-default class-map depends on the action of any ACL that is applied to the policy-map.

The default action can therefore be thought of as specifying the action that will be applied to any data that does not meet the criteria specified by the applied matching commands.

Use the **no** variant of this command to reset to the default action of 'permit'.

Syntax

```
default-action [permit|deny|send-to-cpu|copy-to-cpu|copy-to-mirror|
    send-to-mirror]
```

```
no default-action
```

Parameter	Description
`permit`	Packets to permit.
`deny`	Packets to deny.
`send-to-cpu`	Specify packets to send to the CPU.
`copy-to-cpu`	Specify packets to copy to the CPU.
`copy-to-mirror`	Specify packets to copy to the mirror port.
`send-to-mirror`	Specify packets to send to the mirror port.

Default The default is '`permit`'.

Mode Policy Map Configuration

Examples To set the action for the default class-map to `deny`, use the command:

```
awplus(config-pmap)# default-action deny
```

To set the action for the default class-map to `copy-to-mirror` for use with the mirror interface command, use the command:

```
awplus(config-pmap)# default-action copy-to-mirror
```

Related Commands mirror interface

description (QOS policy map)

Adds a textual description of the policy-map. This can be up to 80 characters long.

Use the **no** variant of this command to remove the current description from the policy-map.

Syntax

```
description <line>
```

```
no description
```

Parameter	Description
<line>	Up to 80 character long line description.

Mode Policy Map Configuration

Example To add the description, VOIP traffic, use the commands:

```
awplus(config-pmap)# description VOIP traffic
```

egress-rate-limit

Sets a limit on the amount of traffic that can be transmitted per second from this port.

Use the **no** variant of this command to disable the limiting of traffic egressing on the interface.

Syntax

```
egress-rate-limit <bandwidth>
```

```
no egress-rate-limit
```

Parameter	Description
<bandwidth>	Bandwidth <1-10000000 kbits per second> (usable units: k, m, g). The egress rate limit can be configured in multiples of 64kbps. If you configure a value that is not an exact multiple of 64kbps, then the value will be rounded up to the nearest higher exact multiple of 64kbps. The minimum is 64 Kb. The default unit is Kb (**k**), but Mb (**m**) or Gb (**g**) can also be specified. The command syntax is not case sensitive, so a value such as 20m or 20M will be taken to mean 20 megabits.

Mode Interface Configuration

Examples To enable egress rate limiting on a port, use the commands:

```
         awplus# configure terminal
 awplus(config)# interface port1.0.1
awplus(config-if)# egress-rate-limit 64k
               % Egress rate limit has been set to 64 Kb
```

To disable egress rate limiting on a port, use the commands:

```
         awplus# configure terminal
 awplus(config)# interface port1.0.1
awplus(config-if)# no egress-rate-limit
```

match access-group

Use this command to define match criterion for a class map.

Syntax

```
match access-group {<hw-IP-ACL>|<hw-MAC-ACL>|<hw-named-ACL>}

no match access-group {<hw-IP-ACL>|<hw-MAC-ACL>|<hw-named-ACL>}
```

Parameter	Description
<hw-IP-ACL>	Specify a hardware IP ACL number in the range <3000-3699>.
<hw-MAC-ACL>	Specify a hardware MAC ACL number in the range <4000-4699>.
<hw-named-ACL>	Specify the hardware named ACL.

Mode Class Map Configuration

Usage First create an access-list that applies the appropriate permit, deny requirements etc. Then use the **match access-group** command to apply this access-list for matching to a class map. Note that this command will apply the access-list matching only to *incoming* data packets.

Examples To configure a class map named cmap1 with one match criterion: access-list 3001, which allows IP traffic from any source to any destination, use the commands:

```
          awplus# configure terminal
  awplus(config)# access-list 3001 permit ip any any
  awplus(config)# class-map cmap1
awplus(config-cmap)# match access-group 3001
```

To configure a class map named cmap2 with one match criterion: access-list 3001, which allows MAC traffic from any source to any destination, use the commands:

```
          awplus# configure terminal
  awplus(config)# access-list 4001 permit any any
  awplus(config)# class-map cmap2
awplus(config-cmap)# match access-group 4001
```

To configure a class map named cmap3 with one match criterion: access-list hw_acl, which allows IP traffic from any source to any destination, use the commands:

```
                  awplus# configure terminal
          awplus(config)# access-list hardware hw_acl
awplus(config-ip-hw-acl)# permit ip any any
          awplus(config)# class-map cmap3
     awplus(config-cmap)# match access-group hw_acl
```

Related Commands class-map

match cos

Sets the CoS for a class-map to match on.

Use the **no** variant of this command to remove CoS.

Syntax

```
match cos <0-7>
```

```
no match cos
```

Parameter	Description
<0-7>	Specify the CoS value.

Mode Class Map Configuration

Examples To set the class-map's CoS to 4, use the commands:

```
              awplus# configure terminal
      awplus(config)# class-map cmap1
 awplus(config-cmap)# match cos 4
```

To remove CoS from a class-map, use the commands:

```
              awplus# configure terminal
      awplus(config)# class-map cmap1
 awplus(config-cmap)# no match cos
```

match dscp

Use this command to define the DSCP to match against incoming packets.

Use the **no** variant of this command to remove a previously defined DSCP.

Syntax

```
match dscp <0-63>

no match dscp
```

Parameter	Description
<0-63>	Specify DSCP value (only one value can be selected).

Mode Class Map Configuration

Usage Use the **match dscp** command to define the match criterion after creating a class map.

Examples To configure a class map named `cmap1` with criterion that matches IP DSCP 56, use the commands:

```
awplus# configure terminal
awplus(config)# class-map cmap1
awplus(config-cmap)# match dscp 56
```

To remove a previously defined DSCP from a class map named cmap1, use the commands:

```
awplus# configure terminal
awplus(config)# class-map cmap1
awplus(config-cmap)# no match dscp
```

Related Commands class-map

match inner-cos

Sets the Inner CoS for a class-map to match on.

Use the **no** variant of this command to remove CoS.

Syntax

```
match inner-cos <0-7>

no match inner-cos
```

Parameter	Description
<0-7>	Specify the Inner CoS value.

Mode Class Map Configuration

Examples To set the class-map's inner-cos to 4, use the commands:

```
             awplus# configure terminal
     awplus(config)# class-map cmap1
awplus(config-cmap)# match inner-cos 4
```

To remove CoS from the class-map, use the commands:

```
             awplus# configure terminal
     awplus(config)# class-map cmap1
awplus(config-cmap)# no match inner-cos
```

match inner-vlan

Use this command to define the inner VLAN ID used as match criteria to classify a traffic class.

Use the **no** variant of this command to disable the VLAN ID used as match criteria.

Syntax

```
match inner-vlan <1-4094>

no match inner-vlan
```

Parameter	Description
<1-4094>	The VLAN number.

Mode Class Map Configuration

Usage This command is used in double-tagged networks to match on a VLAN ID belonging to the client network. For more information on VLAN double-tagged networks, see "Set the Maximum Receive Unit (MRU)" on page 16.5.

Examples To configure a class-map named `cmap1` to include traffic from inner VLAN 3, use the commands:

```
           awplus#  configure terminal
   awplus(config)#  class-map cmap1
awplus(config-cmap)#  match inner-vlan 3
```

To disable the configured VLAN ID as a match criteria for the class-map named `cmap1`, use the commands:

```
           awplus#  configure terminal
   awplus(config)#  class-map cmap1
awplus(config-cmap)#  no match inner-vlan
```

match ip-precedence

Use this command to identify IP precedence values as match criteria.

Use the **no** variant of this command to remove IP precedence values from a class map.

Syntax

```
match ip-precedence <0-7>
```

```
no match ip-precedence
```

Parameter	Description
`<0-7>`	The precedence value to be matched.

Mode Class Map Configuration

Example To configure a class-map named `cmap1` to evaluate all IPv4 packets for a precedence value of 5, use the commands:

```
            awplus# configure terminal
    awplus(config)# class-map cmap1
awplus(config-cmap)# match ip-precedence 5
```

match mac-type

Use this command to set the MAC type for a class-map to match on.

Use **no** variant of this command to remove MAC type.

Syntax

```
match mac-type {l2bcast|l2mcast|l2ucast}
```

```
no match mac-type
```

Parameter	Description
`l2bcast`	Layer 2 Broadcast.
`l2mcast`	Layer 2 Multicast.
`l2ucast`	Layer 2 Unicast.

Mode Class Map Configuration

Examples To set the class-map's MAC type to Layer 2 multicast, use the commands:

```
            awplus# configure terminal
    awplus(config)# class-map cmap1
awplus(config-cmap)# match mac-type l2mcast
```

To remove the class-map's MAC type, use the commands:

```
            awplus# configure terminal
    awplus(config)# class-map cmap1
awplus(config-cmap)# no match mac-type
```

match protocol

Sets the ethernet format and protocol for a class-map to match on. Select one Layer 2 eth-format "and" one Layer 3 protocol.

Use the **no** variant of this command to remove ethernet format and protocol from a class-map.

Syntax

```
match eth-format {<layer-two-format>} protocol {<layer-three-
    protocol>}

no match eth-format protocol
```

Parameter	Description
<layer-two-formats>	
`802dot2-tagged`	802.2 Tagged Packets (enter the parameter name).
`802dot2-untagged`	802.2 Untagged Packets (enter the parameter name).
`ethii-tagged`	EthII Tagged Packets (enter the parameter name).
`ethii-untagged`	EthII Untagged Packets (enter the parameter name).
`netwareraw-tagged`	Netware Raw Tagged Packets (enter the parameter name).
`netwareraw-untagged`	Netware Raw Untagged Packets (enter the parameter name).
`snap-tagged`	SNAP Tagged Packets (enter the parameter name).
`snap-untagged`	SNAP Untagged Packets (enter the parameter name).
<layer-three-protocols>	
`<word>`	A Valid Protocol Number in hexidecimal.
`any`	Note that the parameter "any" is only valid when used with the netwarerawtagged and netwarerawuntagged protocol options.
`sna-path-control`	Protocol Number 04 (enter the parameter name or its number).
`proway-lan`	Protocol Number 0E (enter the parameter name or its number).
`eia-rs Protocol`	Number 4E (enter the parameter name or its number).
`proway Protocol`	Number 8E (enter the parameter name or its number).
`ipx-802dot2`	Protocol Number E0 (enter the parameter name or its number).
`netbeui`	Protocol Number F0 (enter the parameter name or its number).
`iso-clns-is`	Protocol Number FE (enter the parameter name or its number).
`xdot75-internet`	Protocol Number 0801(enter the parameter name or its number).

Parameter(cont.)	Description(cont.)
`nbs-internet`	Protocol Number 0802 (enter the parameter name or its number).
`ecma-internet`	Protocol Number 0803 (enter the parameter name or its number).
`chaosnet`	Protocol Number 0804 (enter the parameter name or its number).
`xdot25-level-3`	Protocol Number 0805 (enter the parameter name or its number).
`arp Protocol`	Number 0806 (enter the parameter name or its number).
`xns-compat`	Protocol Number 0807 (enter the parameter name or its number).
`banyan-systems`	Protocol Number 0BAD (enter the parameter name or its number).
`bbn-simnet`	Protocol Number 5208 (enter the parameter name or its number).
`dec-mop-dump-ld`	Protocol Number 6001 (enter the parameter name or its number).
`dec-mop-rem-cdons`	Protocol Number 6002 (enter the parameter name or its number).
`dec-decnet`	Protocol Number 6003 (enter the parameter name or its number).
`dec-lat`	Protocol Number 6004 (enter the parameter name or its number).
`dec-diagnostic`	Protocol Number 6005 (enter the parameter name or its number).
`dec-customer`	Protocol Number 6006 (enter the parameter name or its number).
`dec-lavc`	Protocol Number 6007 (enter the parameter name or its number).
`rarp`	Protocol Number 8035 (enter the parameter name or its number).
`dec-lanbridge`	Protocol Number 8038 (enter the parameter name or its number).
`dec-encryption`	Protocol Number 803D (enter the parameter name or its number).
`appletalk`	Protocol Number 809B (enter the parameter name or its number).
`ibm-sna`	Protocol Number 80D5 (enter the parameter name or its number).
`appletalk-aarp`	Protocol Number 80F3 (enter the parameter name or its number).

Parameter(cont.)	Description(cont.)
`snmp`	Protocol Number 814Cv.
`ethertalk-2`	Protocol Number 809B (enter the parameter name or its number).
`ethertalk-2-aarp`	Protocol Number 80F3 (enter the parameter name or its number).
`ipx-snap`	Protocol Number 8137 (enter the parameter name or its number).
`ipx-802dot3`	Protocol Number FFFF (enter the parameter name or its number).
`ip`	Protocol Number 0800 (enter the parameter name or its number).
`ipx`	Protocol Number 8137 (enter the parameter name or its number).
`ipv6`	Protocol Number 86DD (enter the parameter name or its number).

Mode Class Map Configuration

Examples To remove the eth-format and protocol from the class-map, use the commands:

```
            awplus# configure terminal
    awplus(config)# class-map cmap1
awplus(config-cmap)# no match eth-format protocol
```

To set the class-map's eth-format to ethii-tagged and protocol to `0800` (IP), use the commands:

```
            awplus# configure terminal
    awplus(config)# class-map
awplus(config-cmap)# match eth-format ethii-tagged protocol
                     0800
```

or

```
awplus(config-cmap)# match eth-format ethii-tagged protocol ip
```

match tcp-flags

Sets one or more tcp flags (control bits) for a class-map to match on.

Use the **no** variant of this command to remove one or more tcp flags for a class-map to match on.

Syntax

```
match tcp-flags {[ack][fin][rst][syn][urg]}

no match tcp-flags {[ack][fin][rst][syn][urg]}
```

Parameter	Description
`ack`	Acknowledge.
`fin`	Finish.
`rst`	Reset.
`syn`	Synchronize.
`urg`	Urgent.

Mode Class Map Configuration

Examples To set the class-map's tcp flags to `ack` and `syn`, use the commands:

```
          awplus# configure terminal
  awplus(config)# class-map
awplus(config-cmap)# match tcp-flags ack syn
```

To remove the tcp-flags `ack` and `rst`, use the commands:

```
          awplus# configure terminal
  awplus(config)# class-map
awplus(config-cmap)# no match tcp-flags ack rst
```

match vlan

Use this command to define the VLAN ID used as match criteria to classify a traffic class.

Use the **no** variant of this command to disable the VLAN ID used as match criteria.

Syntax

```
match vlan <1-4094>

no match vlan
```

Parameter	Description
<1-4094>	The VLAN number.

Mode Class Map Configuration

Examples To configure a class-map named `cmap1` to include traffic from VLAN 3, use the commands:

```
          awplus# configure terminal
  awplus(config)# class-map cmap1
awplus(config-cmap)# match vlan 3
```

To disable the configured VLAN ID as a match criteria for the class-map named `cmap1`, use the commands:

```
          awplus# configure terminal
  awplus(config)# class-map cmap1
awplus(config-cmap)# no match vlan
```

mls qos cos

This command assigns a CoS (Class of Service) user-priority value to untagged frames entering a specified interface. By default, all untagged frames are assigned a CoS value of 0.

Use the **no** variant of this command to return the interface to the default CoS setting for untagged frames entering the interface.

Syntax

```
mls qos cos <0-7>

no mls qos cos
```

Parameter	Description
<0-7>	The Class of Service, user-priority value.

Default By default, all untagged frames are assigned a CoS value of 0. Note that for tagged frames, the default behavior is not to alter the CoS value.

Mode Interface Configuration

Example To assign a CoS user priority value of 3 to all untagged packets entering ports 1.0.1 to 1.0.20, use the commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.1-port1.0.20
awplus(config-if)# mls qos cos 3
```

mls qos enable

Use this command to globally enable QoS on the switch or stack.

Use the **no** variant of this command to globally disable QoS and remove all QoS configuration. The **no** variant of this command removes all class-maps, policy-maps, policers, and queue-sets that have been created. Running the **no mls qos** command will therefore remove all pre-existing QoS configurations on the switch.

Mode Global Configuration

Syntax

```
mls qos enable
```

```
no mls qos
```

Example To enable QoS on the switch, use the commands:

```
        awplus# configure terminal
awplus(config)# mls qos enable
```

mls qos map cos-queue to

Used to set the default CoS to queue mapping. This is the default queue mapping for packets that do not get assigned a queue via any other QoS functionality.

Use the **no** variant of this command to reset the cos-queue map back to its default setting. The default mappings for this command are:

```
        CoS Priority :       0 1 2 3 4 5 6 7
        --------------------------------
        CoS QUEUE:           2 0 1 3 4 5 6 7
```

For more information see, "Mapping CoS tags to traffic types" on page 35.12.

Syntax `mls qos map cos-queue <cos-priority> to <queue-number>`

`no mls qos map cos-queue`

Parameter	Description
<cos-priority>	CoS priority value. Can take a value 0 to 7.
<queue-number>	Queue number. Can take a value 0 to 7.

Mode Global Configuration

Examples To set the cos-queue map back to its defaults, use the command:

```
        awplus# configure terminal
awplus(config)# no mls qos map cos-queue
```

:To map CoS 2 to queue 3, use the command:

```
        awplus# configure terminal
awplus(config)# mls qos map cos-queue 2 to 3
```

Related Commands show mls qos interface

mls qos map premark-dscp to

This command configures the premark-dscp map. It is used when traffic is classified by a class-map that has trust dscp configured. Based on a lookup DSCP, the map determines a new DSCP, COS, queue and bandwidth class for the traffic.

This is used when traffic is classified by a class-map that has trust dscp configured. Based on a lookup DSCP, the map determines a new DSCP, COS, queue and bandwidth class for the traffic.

The **no** variant of this command resets the premark-dscp map to its defaults. If no DSCP is specified then all DSCP entries will be reset to their defaults.

Syntax

```
mls qos map premark-dscp <0-63> to {[new-dscp <0-63>]
   [new-cos <0-7>] [new-bandwidth-class {green|yellow|red}]}
```

```
no mls qos map premark-dscp [<0-63>]
```

Parameter	Description
`premark-dscp <0-63>`	The DSCP value on ingress.
`new-dscp <0-63>`	The DSCP value that the packet will have on egress. If unspecified, this value will remain the DSCP ingress value.
`new-cos <0-7>`	The CoS value that the packet will have on egress. If unspecified, this value will be set to zero.
`new-bandwidth-class`	Modify Egress Bandwidth-class. If unspecified, this value will be set to green.
`green`	Egress Bandwidth-class green (marked down Bandwidth-class).
`yellow`	Egress Bandwidth-class yellow (marked down Bandwidth-class).
`red`	Egress Bandwidth-class red (marked down Bandwidth-class).

Mode Global Configuration

Example To set the entry for DSCP 1 to use a new DSCP of 2, a new CoS of 3, and a new bandwidth class of yellow, use the command:

```
        awplus# configure terminal
awplus(config)# mls qos map premark-dscp 1 to new-dscp 2
                new-cos 3 new-bandwidth-class yellow
```

Related Commands show mls qos maps premark-dscp
trust dscp

no police

Disables any policer previously configured on the class-map.

Syntax

```
no police
```

Mode Priority Map Class Configuration

Usage This command disables any policer previously configured on the class-map.

Example To disable policing on a class-map use the command:

```
           awplus# configure terminal
   awplus(config)# policymap name
awplus(config-pmap)# class classname
awplus(config-pmap-c)# no police
```

Related Commands police single-rate action
police twin-rate action

police single-rate action

Configures a single-rate policer for a class-map.

Syntax

```
police single-rate <cir> <cbs> <ebs> action
    {drop-red|remark-transmit}
```

Parameter	Description	
<*cir*>	Specify the Committed Information Rate (CIR) (1-16000000 kbps).	
<*cbs*>	Specify the Committed Burst Size (CBS) (0-16777216 bytes).	
<*ebs*>	Specify a Excess Burst Size (EBS) (0-16777216 bytes).	
action	Specify the action if rate is exceeded.	
	drop-red	Drop the red packets.
	remark-transmit	Modify the packets using the *remark map*, then transmit. You can configure the remark map using the remark-map command on page 36.30.

Mode Policy Map Class Configuration

Usage A policer can be used to meter the traffic classified by the class-map and as a result will be given one of three bandwidth classes. These are green (conforming), yellow (partially-conforming), and red (non-conforming). A single-rate policer is based on three values. These are the average rate, minimum burst and maximum burst.

Color	Definition
green	The traffic rate is less than the average rate and minimum burst.
yellow	The traffic rate is between the minimum burst and the maximum burst.
red	The traffic rate exceeds the average rate and the maximum burst.

Using an action of drop-red means that any packets classed as red are discarded.

Note This command will not take effect when applied to a class map that attaches to a channel group whose ports span processor instances.

Example To configure a single rate meter measuring traffic of 10 Mbps that drops a sustained burst of traffic over this rate, use the commands:

```
          awplus# configure terminal
  awplus(config)# policy-map name
awplus(config-pmap)# class classname
awplus(config-pmap-c)# police single-rate 10000 1875000 1875000
                   action drop-red
```

Related Commands

no police
police twin-rate action
remark-map

police twin-rate action

Configures a twin-rate policer for a class-map.

Syntax

```
police twin-rate <cir> <pir> <cbs> <pbs> action
    {drop-red|remark-transmit}
```

Parameter	Description	
<cir>	Specify the Committed Information Rate (CIR) (1-16000000 kbps).	
<pir>	Specify the Peak Information Rate (PIR) (kbps).	
<pbs>	Specify the Peak Burst Size (PBS) (0-16777216 bytes).	
action	Specify the action if rate is exceeded.	
	drop-red	Drop the red packets.
	remark-transmit	Modify the packets using the *remark map*, then transmit. You can configure the remark map using the remark-map command on page 36.30.

Mode Policy Map Class Configuration

Usage A policer can be used to meter the traffic classified by the class-map and as a result will be given one of three bandwidth classes. These are green (conforming), yellow (partially-conforming), and red (non-conforming).

A twin-rate policer is based on four values. These are the minimum rate, minimum burst size, maximum rate, and maximum burst size.

Bandwidth Class	Definition
green	The sum of the number of existing (buffered) bytes plus those arriving at the port per unit time, result in a value that is less than that set for the CBS.
yellow	The sum of the number of existing (buffered) bytes plus those arriving at the port per unit time, result in a value that is between those set for the CBS and the PBS.
red	The sum of the number of existing (buffered) bytes plus those arriving at the port per unit time, result in a value that exceeds that set for the PBS.

Using an action of drop-red means that any packets classed as red will be discarded.

When using an action of remark-transmit the packet will be remarked with the values configured in the policed-dscp map. The index into this map is determined by the DSCP in the packet.

Example

To configure a twin rate meter measuring a minimum rate of 10 Mbps and a maximum rate of 20 Mbps that uses the premark map to remark any non-conforming traffic, use the commands:

```
           awplus# configure terminal
   awplus(config)# policy-map name
awplus(config-pmap)# class classname
awplus(config-pmap-c)# police twin-rate 10000 20000 1875000
                       3750000 action remark-transmit
```

Related Commands

no police
police twin-rate action

policy-map

Use this command to create a policy map and to enter Policy Map Configuration mode to configure the specified policy map.

Use the **no** variant of this command to delete an existing policy map.

Syntax

```
policy-map <name>

no policy-map <name>
```

Parameter	Description
<*name*>	Name of the policy map.

Mode Global Configuration

Example To create a policy-map called `pmap1`, use the commands:

```
          awplus#  configure terminal
   awplus(config)#  policy-map pmap1
awplus(config-pmap)#
```

Related Commands class-map

priority-queue

Configures strict priority based scheduling on the specified egress queues. You must specify at least one queue.

Syntax

```
priority-queue {0}[1][2][3][4][5][6][7]
```

Parameter	Description
`{0}[1]...[7]`	Specify queues to apply the scheduling rule in the range <0-7>. To apply priorities of 4, 5 and 6 to queues 0, 1 and 2 enter: `awplus(config-if)#priority-queue 4 5 6`

Mode Interface Configuration

Usage By default, the queues on all ports are set for priority queueing. You can change the queue emptying sequence to weighted round robin, by using the wrr-queue weight queues command on page 36.53: You can then use the priority-queue command to reset the selected queues to priority queueing. Note that the emptying sequence for priority queueing is always highest queue number to lowest queue number.

For more information on queueing operation, see the chapter, Quality of Service (QoS) Introduction.

Example To apply priority based scheduling of 5, 6 and 7 to to egress queues 0, l and 2, use the commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.1.12
awplus(config-if)# priority-queue 5 6 7
```

Related Commands show mls qos interface
show mls qos interface queue-counters
wrr-queue weight queues

remark-map

Configures the remark map. This command is applied when a policer is configured with the **action** parameter of the command, police single-rate action set to **remark-transmit**.

The **no** variant of this command resets the remark map to its defaults. Specifying the bandwidth class is optional. If no bandwidth class is specified, then all bandwidth classes are reset to their defaults.

Syntax

```
remark-map [bandwidth-class {green|yellow|red}] to {[new-dscp <0-63>]
    [new-bandwidth-class {green|yellow|red}]}
```

```
no remark-map [bandwidth-class {green|yellow|red}] to {[new-dscp
    <0-63>] [new-bandwidth-class {green|yellow|red}]}
```

Parameter	Description
`bandwidth-class`	Specify the bandwidth class of packets to remark.
`green`	Remark green packets.
`yellow`	Remark yellow packets.
`red`	Remark red packets.
`new-dscp`	Specify the new dscp value.
<0-63>	The DSCP value.
`new-bandwidth-class`	Specify the new bandwidth class.
`green`	Remark the packet green.
`yellow`	Remark the packet yellow.
`red`	Remark the packet red.

Mode Policy Map Class Configuration

Examples To remark the policed green traffic to a new DSCP of 2 and a new bandwidth class of `yellow`, use the commands:

```
          awplus# configure terminal
  awplus(config)# policy-map pmap1
awplus(config-pmap)# class cmap1
awplus(config-pmap-c)# remark-map bandwidth-class green to
                       new-dscp 2 new-bandwidth-class yellow
```

To reset the DSCP for all bandwidth classes, use the commands:

```
           awplus# configure terminal
   awplus(config)# policy-map pmap1
awplus(config-pmap)# class cmap1
awplus(config-pmap-c)# no remark-map to new dscp
```

Related Commands police single-rate action
police twin-rate action

remark new-cos

Enables you to configure and remark either or both the CoS flag in the data packet, and the input into the CoS to queue map thus changing the destination egress queue.

Syntax

```
remark new-cos <0-7> [internal|external|both]
```

```
no remark new-cos [internal|external|both]
```

Parameter	Description
`<0-7>`	The new value for either the CoS flag or the input into the CoS to queue map.
`external`	Remarks the CoS flag in the packet.
`internal`	Remarks the new-CoS input into the CoS to queue map.
`both`	Remarks (with the same value) both the CoS flag in the packet and the input to the CoS to queue map.

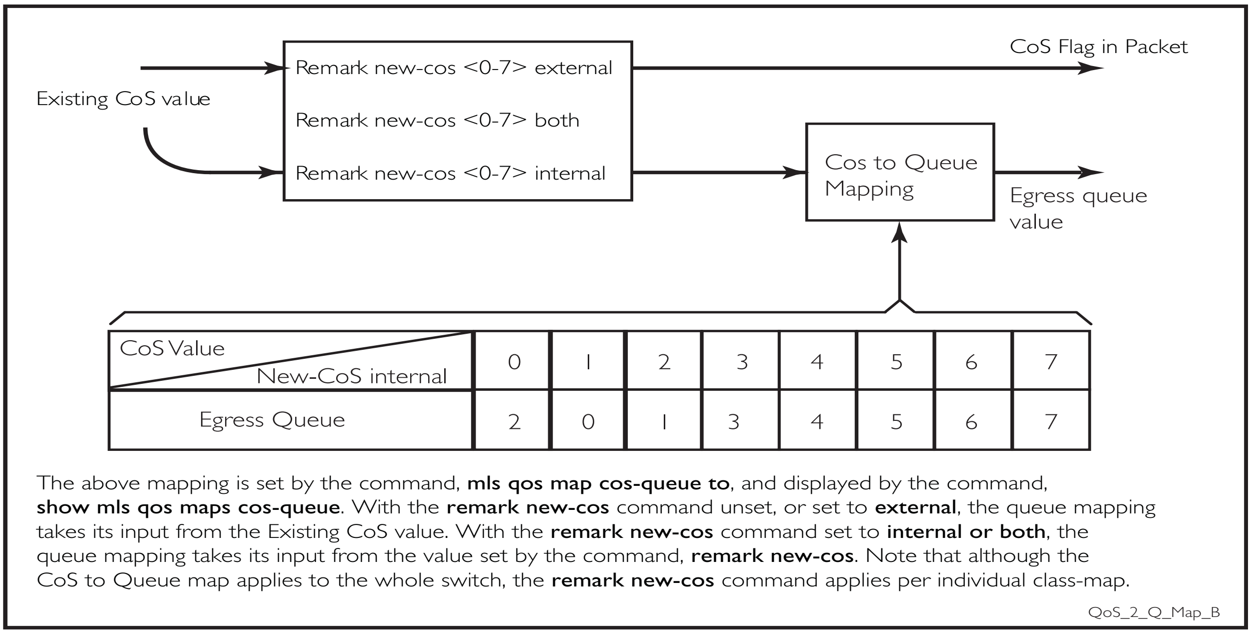
Mode Policy Map Class Configuration

Usage The default CoS to Queue mappings are shown in the following table:

CoS Value	0	1	2	3	4	5	6	7
Egress Queue No	2	0	1	3	4	5	6	7

The relationship between this command and the CoS to queue map is shown in Figure 36-1.

Figure 36-1: Remarking and the CoS to Q Map


CoS Flag in Packet
Existing CoS value
Remark new-cos <0-7> external
Remark new-cos <0-7> both
Remark new-cos <0-7> internal
Cos to Queue Mapping
Egress queue value

CoS Value / New-CoS internal	0	1	2	3	4	5	6	7
Egress Queue	2	0	1	3	4	5	6	7

The above mapping is set by the command, **mls qos map cos-queue to**, and displayed by the command, **show mls qos maps cos-queue**. With the **remark new-cos** command unset, or set to **external**, the queue mapping takes its input from the Existing CoS value. With the **remark new-cos** command set to **internal or both**, the queue mapping takes its input from the value set by the command, **remark new-cos**. Note that although the CoS to Queue map applies to the whole switch, the **remark new-cos** command applies per individual class-map.

QoS_2_Q_Map_B

Table 36-1: CoS to Egress Queue Remarking Function

Input	Command	Output
CoS field = 1	Remark new-cos (not configured)	CoS value = 1 Packet sent to egress queue 0
CoS field = 1	Remark new-cos 2 external	CoS value = 2 Packet sent to egress queue 0
CoS set to 1	Remark new-cos 2 internal	CoS value = 1 Packet sent to egress queue 1
CoS set to 1	Remark new-cos 2 both	CoS value = 2 Packet sent to egress queue 1
Note: This table assumes that the CoS to Queue map is set to its default values.		

Example For policy-map `pmap3` and class-map `cmap1`, set the CoS value to 2 and also set the input to the CoS to queue map so that the traffic is assigned to egress queue 1:

```
awplus# configure terminal
awplus(config)# policy-map pmap3
awplus(config-pmap)# class cmap1
awplus(config-pmap-c)# remark new-cos 2 both
```

Related Commands mls qos map cos-queue to
show mls qos maps cos-queue

service-policy input

Use this command to apply a policy map to the input of an interface.

Use the **no** variant of this command to remove a policy map and interface association.

Syntax

```
service-policy input <policy-map>
no service-policy input <policy-map>
```

Parameter	Description
<policy-map>	Policy map name that the input will applied to.

Mode Interface Configuration

Usage This command can be applied to switch ports or static channel groups, but not to dynamic (LACP) channel groups.

Example To apply a policy map named pmap1 to interface port1.0.2, use the commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.2
awplus(config-if)# service-policy input pmap1
```

set ip next-hop (PBR)

Forwards traffic matching this class map to the specified nexthop.

When this command is set, all packets that match a selected class map will be forwarded to the specified nexthop.

The **no** variant of this command removes the next-hop address (in the context of its policy-map and class-map) from the configuration.

Syntax

```
set ip next-hop <ip-addr>

no set ip next-hop
```

Parameter	Description
<ip-addr>	The IP address of the next hop destination.

Mode Policy Map Class Configuration

Usage In typical deployments of policy-based routing, some traffic types require normal routing (i.e. via the routes in the IP routing table) while other traffic types require policy based routing.

Where the traffic to be policy routed is a subset of the traffic that is to be normally routed, then the configuration is reasonably simple. The policy-map will contain one or more classes that match the traffic to be policy routed, and will have their next-hop configured by this command - **set ip next-hop (PBR).** The remaining traffic will be conventionally routing routed according to the rules set for the default class - providing that this is not subject to the **set ip next-hop (PBR).**

The situation becomes a little more complex where the traffic requiring normal routing is a subset of the traffic to be policy-routed. In this situation the policy-map would need to contain one, or more, classes that match the requirement for normal routing, These classes would not be configured with a **set ip next-hop (PBR)** command. Then the remaining traffic classes that require normal routing would have the **set ip next-hop (PBR)** command applied to them. Note that this traffic could be just the default class, if ALL other traffic types were to be policy-routed.

Also note that the order in which the classes are configured in the policy-map is important; because traffic is matched against the classes in the order that they were assigned to the policy map.

Details of a practical example of such a policy-based routing is shown in "Policy-Based Routing" on page 35.24.

Example To forward a packet to a 192.168.1.1, use the commands:

```
            awplus#  configure terminal
    awplus(config)#  policy-map pmap1
awplus(config-pmap)#  class cmap1
awplus(config-pmap-c)#  set ip next-hop 192.168.1.1
```

show class-map

Use this command to display the QoS class maps to define the match criteria to classify traffic.

Syntax `show class-map <class-map name>`

Parameter	Description
<class-map name>	Name of the class map.

Mode User Exec and Privileged Exec

Example To display the QoS class maps to define the match criteria to classify traffic, use the command:

```
awplus# show class-map cmap1
```

Output Figure 36-2: Example output from the **show class-map** command

```
CLASS-MAP-NAME: cmap1
      Set IP DSCP: 56
      Match IP DSCP: 7
```

Related Commands class-map

show mls qos interface

Displays the current settings for the interface. This includes it's default CoS and queue, scheduling used for each queue, and any policies/maps that are attached.

Syntax `show mls qos interface [<port>]`

Parameter	Description
<port>	Switch port.

Mode User Exec and Privileged Exec

Example To display current CoS and queue settings for interface `port1.0.1`, use the command:

```
awplus# show mls qos interface port1.0.1
```

Output Figure 36-3: Example output from the **show mls qos interface** command

```
Default CoS:   7
    Default Queue: 7
    Number of egress queues: 8
    Queue Set: 1
    Egress Queue:        0
      Status:            Enabled
      Scheduler:         Strict Priority
      Queue Limit:       12%
      Egress Rate Limit: 0 Kb
    Egress Queue:        1
      Status:            Enabled
      Scheduler:         Strict Priority
      Queue Limit:       12%
      Egress Rate Limit: 0 Kb
    Egress Queue:        2
      Status:            Enabled
      Scheduler:         Strict Priority
      Queue Limit:       12%
      Egress Rate Limit: 0 Kb
    Egress Queue:        3
      Status:            Enabled
      Scheduler:         Wrr Group 2
      Weight:            10
      Queue Limit:       12%
      Egress Rate Limit: 0 Kb
    Egress Queue:        4
      Status:            Enabled
      Scheduler:         Wrr Group 1
      Weight:            10
      Queue Limit:       12%
      Egress Rate Limit: 0 Kb
    Egress Queue:        5
      Status:            Enabled
      Scheduler:         Strict Priority
      Queue Limit:       12%
      Egress Rate Limit: 0 Kb
    Egress Queue:        6
      Status:            Enabled
      Scheduler:         Strict Priority
      Queue Limit:       12%
      Egress Rate Limit: 0 Kb
    Egress Queue:        7
      Status:            Enabled
      Scheduler:         Strict Priority
      Queue Limit:       12%
      Egress Rate Limit: 0 Kb
```

Table 36-2: Parameters in the output of the **show mls qos interface** command

Parameter	Description
`Default CoS`	The default CoS priority that will be applied to all packets arriving on this interface.
`Default Queue`	The default queue that will be applied to all packets arriving on this interface.
`Number of egress queues`	The total number of egress queues available on this interface.
`Queue Set`	Drop queue set that has been applied to the port. This could either be operating in threshold or random-detect mode.
`Egress Queue X`	Number of this egress queue.
`Status`	Queue can either be enabled or disabled.
`Scheduler`	The scheduling mode being used for servicing the transmission of packets on this port.
`Queue Limit`	The percentage of the ports buffers that have been allocated to this queue.
`Egress Rate Limit`	The amount of traffic that can be transmitted via this queue per second. 0 Kb means there is currently no rate-limiting enabled.

show mls qos interface policer-counters

Display an interface's policer counters. This can either be for a specific class-map or for all class-maps attached to the interface. If no class-map is specified all class-map policer counters attached to the interface will be displayed.

These are the counters based on metering performed on the specified class-map. Therefore, the 'Dropped packets' counter is the number of bytes dropped due to metering. This is different from the packets dropped via a 'deny' action in the ACL.

You must enable the QoS counter platform enhanced mode before running this command.

Syntax

```
show mls qos interface <port> policer-counters
    [class-map <class-map>]
```

Parameter	Description
`<port>`	Switch port.
`class-map`	Select a class-map.
`<class-map>`	Class-map name.

Mode User Exec and Privileged Exec

Example To show the counters for all class-maps attached to `port1.0.1`, use the command:

```
awplus# show mls qos interface port1.0.1 policer-counters
```

To show the counters for all class-maps attached to `port1.1.1`, use the command:

```
awplus# show mls qos interface port1.1.1 policer-counters
```

Output Figure 36-4: Example output from the **show mls qos interface policer-counters** command

```
Interface:           port1.0.1
   Class-map:          cmap1
     Aggregate Bytes: 128
     Red Bytes:       0
```

show mls qos interface queue-counters

Display an interface's egress queue counters. This can either be for a specific queue or for all queues on the interface. If no queue is specified all queue counters on the interface will be displayed.

The counters show the number of frames currently in the queue and the maximum number of frames allowed in the queue, for individual egress queues and the port's queue (which will be a sum of all eight egress queues).

Syntax

```
show mls qos interface <port> queue-counters queue [<0-7>]
```

Parameter	Description
<port>	Switch port.
<0-7>	Queue.

Mode User Exec and Privileged Exec

Example To show the counters for all queues on `port1.0.1`, use the command:

```
awplus# show mls qos interface port1.0.1 queue-counters
```

Output Figure 36-5: Example output from the **show mls qos interface queue-counters** command

```
Interface port1.0.1 Queue Counters:
   Port queue length        0 (maximum 896)
   Egress Queue length:
     Queue 0                0 (maximum 112)
     Queue 1                0 (maximum 112)
     Queue 2                0 (maximum 112)
     Queue 3                0 (maximum 112)
     Queue 4                0 (maximum 112)
     Queue 5                0 (maximum 112)
     Queue 6                0 (maximum 112)
     Queue 7                0 (maximum 112)
```

Table 36-3: Parameters in the output of the **show mls qos interface queue-counters** command

Parameter	Description
`Interface`	Port we are showing the counters for.
`Port queue length`	Number of frames in the port's queue. This will be the sum of all egress queues on the port.
`Egress Queue length`	Number of frames in a specific egress queue.

show mls qos interface storm-status

Show the current configuration and status of the QoS Storm Protection (QSP) on the given port.

Syntax `show mls qos interface <port> storm-status`

Parameter	Description
<port>	Switch port.

Mode User Exec and Privileged Exec

Example To see the QSP status on `port1.0.1`, use command:

```
awplus# show mls qos interface port1.0.1 storm-status
```

Output Figure 36-6: Example output from the **show mls qos interface storm-status** command

```
Interface:           port1.0.1
Storm-Protection:    Enabled
Port-status:         Enabled
Storm Action:        vlandisable
Storm Window:        5000 ms
Storm Downtime:      0 s
Timeout Remaining:   0 s
Last read data-rate: 0 kbps
Storm Rate:          1000 kbps
```

Related Commands storm-action
storm-downtime
storm-protection
storm-rate
storm-window

show mls qos maps cos-queue

Show the current configuration of the cos-queue map.

Syntax `show mls qos maps cos-queue`

Mode User Exec and Privileged Exec

Example To display the current configuration of the cos-queue map, use the command:

```
awplus# show mls qos maps cos-queue
```

Output Figure 36-7: Example output from the **show mls qos maps cos-queue** command

```
COS-TO-QUEUE-MAP:
      COS :           0 1 2 3 4 5 6 7
      --------------------------------
      QUEUE:          0 7 1 3 4 5 6 7
```

Related Commands mls qos map cos-queue to

show mls qos maps premark-dscp

Displays the premark-dscp map. This map is used when the trust dscp command has been specified for a policymap's class-map to replace the DSCP, CoS, queue, and bandwidth class of a packet matching the class-map based on a lookup DSCP value.

Syntax

```
show mls qos maps premark-dscp [<0-63>]
```

Parameter	Description
<0-63>	DSCP table entry.

Mode User Exec and Privileged Exec

Example To display the premark-dscp map for DSCP 1, use the command:

```
awplus# show mls qos maps premark-dscp 1
```

Output Figure 36-8: Example output from the **show mls qos maps premark-dscp** command

```
PREMARK-DSCP-MAP:

    DSCP 1
    Bandwidth Class        Green    Yellow   Red
    -------------------------------------------------
    New DSCP               1        -        -
    New CoS                0        -        -
    New Queue              0        -        -
    New Bandwidth Class    green    -        -
```

Related Commands mls qos map premark-dscp to
trust dscp

show policy-map

Displays the policy-maps configured on the switch. The output also shows whether or not they are connected to a port (attached / detached) and shows their associated class-maps.

Syntax `show policy-map [<name>]`

Parameter	Description
<name>	The name of a specific policy map.

Mode User Exec and Privileged Exec

Example To display a listing of the policy-maps configured on the switch, use the command:

```
awplus# show policy-map
```

Output Figure 36-9: Example output from the **show policy-map** command

```
POLICY-MAP-NAME: general-traffic
  State: attached
    Default class-map action: permit
    CLASS-MAP-NAME: default
    CLASS-MAP-NAME: database-traffic
```

Related Commands service-policy input

storm-action

Sets the action to take when triggered by QoS Storm Protection (QSP). There are three available options:

- **portdisable** will disable the port in software.
- **vlandisable** will disable the port from the VLAN matched by the class-map in class-map.
- **linkdown** will physically bring the port down. The **vlandisable** requires the match vlan class-map to be present in the class-map.

The **no** variant of this command will negate the action set by the **storm-action** command.

Syntax

```
storm-action {portdisable|vlandisable|linkdown}
```

```
no storm-action
```

Parameter	Description
`portdisable`	Disable the port in software.
`vlandisable`	Disable the VLAN.
`linkdown`	Shutdown the port physically.

Mode Policy Map Class Configuration

Examples To apply the storm protection of `vlandisable` to the policy map named `pmap2`, and the class-map named `cmap1`, use the following commands:

```
          awplus# configure terminal
  awplus(config)# policy-map pmap2
awplus(config-pmap)# class cmap1
awplus(config-pmap-c# storm-action vlandisable
```

To negate the storm protection set on the policy map named `pmap2`, and the class-map named `cmap1`, use the following commands:

```
          awplus# configure terminal
  awplus(config)# policy-map pmap2
awplus(config-pmap)# class cmap1
awplus(config-pmap-c# no storm-action
```

Related Commands

storm-downtime
storm-protection
storm-rate
storm-window

storm-downtime

Sets the time to re-enable the port once disabled by QoS Storm Protection (QSP). The time is given in seconds, from a minimum of one second to maximum of 86400 seconds (i.e. one day).

The **no** variant of this command resets the time to the default value of 10 seconds.

Syntax

```
storm-downtime <1-86400>

no storm-downtime
```

Parameter	Description
<1-86400>	Seconds.

Default 10 seconds

Mode Policy Map Class Configuration

Examples To re-enable the port in 1 minute, use the following commands:

```
          awplus# configure terminal
  awplus(config)# policy-map pmap2
awplus(config-pmap)# class cmap1
awplus(config-pmap-c)# storm-downtime 60
```

To re-set the port to the default (10 seconds), use the following commands:

```
          awplus# configure terminal
  awplus(config)# policy-map pmap2
awplus(config-pmap)# class cmap1
awplus(config-pmap-c)# no storm-downtime
```

Related Commands

storm-action
storm-protection
storm-rate
storm-window

storm-protection

Use this command to enable the Policy Based Storm Protection (such as QSP - QoS Storm Protection). Storm protection is activated as soon as a port is enabled.

The **no** variant of this command disables Policy Based Storm Protection.

Syntax

```
storm-protection

no storm-protection
```

Default By default, storm protection is disabled.

Mode Policy Map Class Configuration

Examples To enable QSP on `cmap2` in `pmap2`, use the following commands:

```
            awplus# policy-map pmap2
  awplus(config-pmap)# class cmap2
awplus(config-pmap-c)# storm-protection
```

To disable QSP on `cmap2` in `pmap2`, use the following commands:

```
            awplus# policy-map pmap2
  awplus(config-pmap)# class cmap2
awplus(config-pmap-c)# no storm-protection
```

Related Commands

storm-action
storm-downtime
storm-rate
storm-window

storm-rate

Sets the data rate that triggers the storm-action. The rate is in kbps and the range is from 1kbps to 10Gbps.

Note that this setting is made in conjunction with the **storm window** command.

Use the **no** variant of this command to negate the **storm-rate** command.

Syntax

```
storm-rate <1-10000000>
```

```
no storm-rate
```

Parameter	Description
<1-10000000>	The range of the storm-rate.

Default No default

Mode Policy Map Class Configuration

Usage This setting is made in conjunction with the storm-window command on page 36.49.

Examples To the limit to 1Mbps, use the following commands:

```
             awplus#  configure terminal
     awplus(config)#  policy-map pmap2
awplus(config-pmap)#  class cmap2
awplus(config-pmap-c)#  storm-rate 1000
```

To negate the limit set previously, use the following commands:

```
             awplus#  configure terminal
     awplus(config)#  policy-map pmap2
awplus(config-pmap)#  class cmap2
awplus(config-pmap-c)#  storm-rate 1000
```

Related Commands storm-action
storm-downtime
storm-protection
storm-window

storm-window

Sets the window size of QoS Storm Protection (QSP). This sets the time to poll the data-rate every given milliseconds. Minimum window size of 100 ms and the maximum is 60 sec.

Use the **no** variant of this command to negate the **storm-window** command.

Syntax

```
storm-window <100-60000>

no storm-window
```

Parameter	Description
<100-60000>	The window size, measured in milliseconds.

Default No default

Mode Policy Map Class Configuration

Usage This command should be set in conjunction with the storm-rate command on page 36.48.

Examples To set the QSP window size to 5000 ms, use the following commands:

```
              awplus#  configure terminal
       awplus(config)#  policy-map pmap2
  awplus(config-pmap)#  class cmap2
awplus(config-pmap-c)#  storm-window 5000
```

To negate the QSP window size set previously, use the following commands:

```
              awplus#  configure terminal
       awplus(config)#  policy-map pmap2
  awplus(config-pmap)#  class cmap2
awplus(config-pmap-c)#  storm-window 5000
```

Related Commands

storm-action
storm-downtime
storm-protection
storm-rate

trust dscp

Use this command to enable the premark-dscp map to replace the bandwidth-class, cos, dscp, and queue of classified traffic within a policy-map based on a lookup DSCP value.

With the **no** variant of this command, no premark-dscp mapping function will be applied for the selected policy-map. QoS components of the packet existing at ingress will pass unchanged.

Syntax

```
trust dscp
```

```
no trust
```

Mode Policy Map Configuration

Examples To enable the premark-dscp map lookup for policy-map `pmap1`, use the commands:

```
           awplus#  configure terminal
   awplus(config)#  policy-map pmap1
awplus(config-pmap)#  trust dscp
```

To disable the premark-dscp map lookup for policy-map `pmap1`, use the commands:

```
           awplus#  configure terminal
   awplus(config)#  policy-map pmap1
awplus(config-pmap)#  no trust
```

Related Commands mls qos map premark-dscp to

wrr-queue disable queues

Use this command to disable an egress queue from transmitting traffic.

The **no** variant of this command enables an egress queue to transmit traffic.

Syntax

```
wrr-queue disable queues [0][1][2][3][4][5][6][7]

no wrr-queue disable queues [0][1][2][3][4][5][6][7]
```

Parameter	Description
`[1][2]...[7]`	Selects one or more queues numbered 0 to 7.

Mode Interface Configuration

Examples To enable queues 1-3 to transmit traffic, use the commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.1
awplus(config-if)# no wrr-queue disable queues 1 2 3
```

To disable queues 1-3 from transmitting traffic, use the commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.1
awplus(config-if)# wrr-queue disable queues 1 2 3
```

Related Commands show mls qos interface

wrr-queue egress-rate-limit queues

Sets a limit on the amount of traffic that can be transmitted per second from these queues. The default unit is in Kb, but Mb or Gb can also be specified. The minimum is 651Kb.

Syntax

```
wrr-queue egress-rate-limit <bandwidth> queues
    {0}[1][2][3][4][5][6][7]

no wrr-queue egress-rate-limit <bandwidth> queues
    {0}[1][2][3][4][5][6][7]
```

Parameter	Description
`<bandwidth>`	Bandwidth <1-10000000 kbits> (usable units: k, m, g).
`{0}[1]...[7]`	Selects one or more queues to apply the bandwidth limit to as specified in the preceding `<bandwidth>` parameter. Then apply a priority in the range <0-7> for each selected queue. For example, to apply priorities 1, 2 and 3 to queues 0, 1 and 2 enter queues 1 2 3. {0} [1] . . . [7] indicates a queue for a priority <0-7>.

Mode Interface Configuration

Example To set enable egress rate limiting on queues 0, 1 and 2 with the priorities 1, 2 and 3, use the commands:

```
          awplus#  configure terminal
  awplus(config)#  interface port1.0.1
awplus(config-if)#  wrr-queue egress-rate-limit 500M
                    queues 1 2 3
```

Related Commands show mls qos interface

wrr-queue weight queues

Configures weighted round-robin based scheduling on the specified egress queues on switch port interfaces only. The weights are specified as ratio's relative to each other.

Syntax

```
wrr-queue weight <1-15> queues {0}[1][2]3][4][5][6][7]
```

Parameter	Description
<1-15>	Weight (the higher the number the greater will be the queue servicing).
{0}[1]...[7]	Egress queues 0-7 to select and assign a priority in the range <0-7>. The queue number is indicated by the order of entry. For example, queue 1 2 assigns priority 1 and 2 to queues 0 and 1 due to the order of entry. Queue 0 is a required queue.

Mode Interface Configuration for switch port interfaces only (not for static aggregated interfaces).

Usage Only apply weighted round-robin based scheduling to switch port interfaces (for example, `awplus(config)#interface port1.0.2`).

You cannot apply weighted round-robin based scheduling to static aggregated interfaces (for example, `awplus(config)#interface sa2`). Attempting to applying weighted round-robin based scheduling on aggregated interfaces will display the console error shown below:

```
awplus(config)# interface sa2
awplus(config-if)# wrr-queue weight
                                 ^
% Invalid input detected at ^ marker
```

Examples To apply a wrr weight of 6 to queues `0 1 2`, on `port1.0.1`, use the commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.1
awplus(config-if)# wrr-queue weight 6 queues 0 1 2
```

To share the bandwidth on `port1.0.1` between queues `1` to `3` such that queue `2` gets serviced twice as often as queue `1` and queue `3` gets serviced twice as often as queue `2`, use the commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.1
awplus(config-if)# wrr-queue weight 1 queues 1
awplus(config-if)# wrr-queue weight 2 queues 2
awplus(config-if)# wrr-queue weight 3 queues 4
```

Related Commands priority-queue
show mls qos interface

Chapter 37: 802.1X Introduction and Configuration

Introduction

The IEEE Standard 802.1X provides a method of restricting access to networks based on authentication information. 802.1X provides port-based network access control for devices connected to the Ethernet. This allows a network controller to restrict external devices from gaining access to the network behind an 802.1X controlled port. External devices that wish to access services via a port under 802.1X control must firstly authenticate themselves and gain authorization before any packets originating from, or destined for, the external device are allowed to pass through the 802.1X controlled port.

The 802.1X Implementation

802.1X port access control is achieved by making devices attached to a controlled port authenticate themselves via communication with an authentication server before these devices are allowed to access the network behind the controlled port.

Authentication is required on a per-port basis. The main components of an 802.1X implementation are:

- the authenticator - the port on this device that wishes to enforce authentication before allowing access to services that are accessible behind it.
- the supplicant - the port that wishes to access services offered by the authenticator's system. The supplicant may be a port on a PC or other device connected to this device.
- the authentication server - a device that uses the authentication credentials supplied by the supplicant, via the authenticator, to determine if the authenticator should grant access to its services.

Configuring 802.1X

The following example explains how to configure 802.1X. In this example, the RADIUS Server keeps the Client information, validating the identity of the Client and updating the switch about the authentication status of the client. The switch is the physical access between the two clients and the server. It requests information from the client, relays information to the server and then back to the client.

To configure 802.1X authentication, first enable authentication on `port1.0.1` and `port1.0.2` and then specify the RADIUS Server IP address and port.

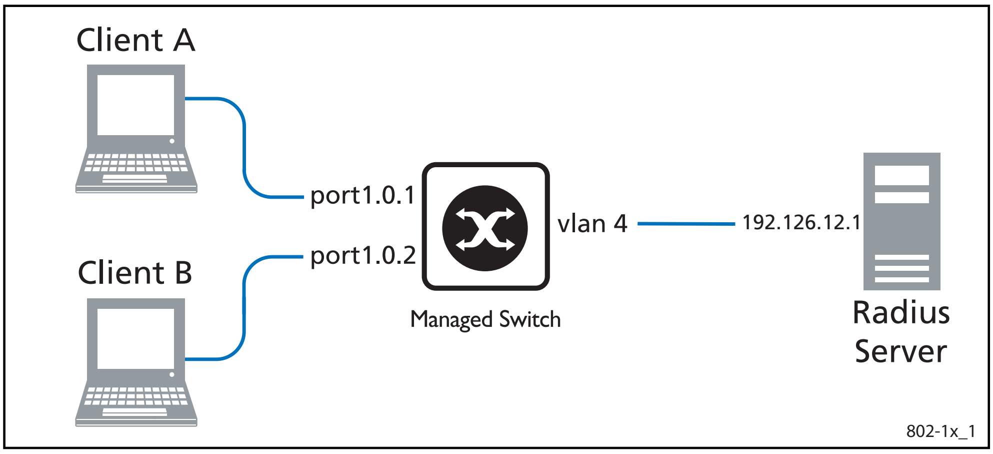


Table 37-1: 802.1X configuration on the switch

awplus# configure terminal	Enter the Global Configuration mode.
awplus(config)# aaa authentication dot1x default group radius	Enable authentication globally.
awplus(config)# interface port1.0.1	Specify the interface (port1.0.1) to be configured and enter the Interface mode.
awplus(config-if)# dot1x port-control auto	Enable authentication (via RADIUS) on port1.0.1.
awplus(config-if)# dot1x control-direction both	Block traffic in both directions, other than authentication packets, until authentication is complete.
awplus(config-if)# exit	Exit the Interface Configuration mode and enter the Global Configuration mode.
awplus(config)# interface port1.0.2	Specify the interface (port1.0.2) you are configuring and enter the Interface mode.
awplus(config-if)# dot1x port-control auto	Enable authentication (via RADIUS) on port1.0.2.
awplus(config-if)# exit	Exit the Interface Configuration mode and enter the Global Configuration mode.
awplus(config)# radius-server host 192.126.12.1 auth-port 1812	Specify the RADIUS Server address (192.126.12.1) and authentication port.
awplus(config)# radius-server key secret	Specify the shared key secret between the RADIUS server and the client.
awplus(config)# interface vlan4	Specify the vlan (vlan4) to be configured and enter the Interface mode.
awplus(config-if)# ip address 192.126.12.2/24	Set the IP address on vlan4.

Names of Commands Used

dot1x port-control
radius-server host
radius-server key

Validation Commands

show dot1x
show dot1x interface

Chapter 38: 802.1X Commands

Command List

This chapter provides an alphabetical reference of commands used to configure 802.1X port access control. For more information, see Chapter 37, 802.1X Introduction and Configuration.

debug dot1x

Use this command to enable 802.1X IEEE Port-Based Network Access Control troubleshooting functions.

Use the **no** variant of this command to disable this function.

Syntax

```
debug dot1x [all|auth-web|event|nsm|packet|timer]

no debug all dot1x

no debug dot1x [all|auth-web|event|nsm|packet|timer]
```

Parameter	Description
`all`	Used with the **no** variant of this command exclusively; turns off all debugging for 802.1X.
`auth-web`	Specifies debugging for 802.1X auth-web information.
`events`	Specifies debugging for 802.1X events.
`nsm`	Specifies debugging for NSM messages.
`packet`	Specifies debugging for 802.1X packets.
`timer`	Specifies debugging for 802.1X timers.

Mode Privileged Exec and Global Configuration

Usage This command without any parameters turns on normal 802.1X debug information.

```
awplus# debug dot1x
```

```
awplus# show debugging dot1x
```

```
802.1X debugging status:
  802.1X events debugging is
  802.1X timer debugging is on
  802.1X packets debugging is on
  802.1X NSM debugging is on
```

Examples

```
awplus# debug dot1x

awplus# debug dot1x all
```

Related Commands show debugging dot1x
undebug dot1x

dot1x control-direction

This command sets the direction of the filter for the unauthorized interface.

If the optional **in** parameter is specified with this command then packets entering the specified port are discarded. The **in** parameter discards the ingress packets received from the supplicant.

If the optional **both** parameter is specified with this command then packets entering (ingress) and leaving (egress) the specified port are discarded. The **both** parameter discards the packets received from the supplicant and sent to the supplicant.

The **no** variant of this command sets the direction of the filter to **both**. The port will then discard both ingress and egress traffic.

Syntax

```
dot1x control-direction {in|both}

no dot1x control-direction
```

Parameter	Description
`in`	Discard received packets from the supplicant (ingress packets).
`both`	Discard received packets from the supplicant (ingress packets) and transmitted packets to the supplicant (egress packets).

Default The authentication port direction is set to **both** by default.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

Example To set the port direction to the default (**both**) for `port1.0.2`, use the following commands:

```
          awplus#  configure terminal
  awplus(config)#  interface port1.0.2
awplus(config-if)#  no dot1x control-direction
```

To set the port direction to **in** for `port1.0.2`, use the following commands:

```
          awplus#  configure terminal
  awplus(config)#  interface port1.0.2
awplus(config-if)#  dot1x control-direction in
```

Validation Commands

show dot1x
show dot1x interface
show auth-mac interface
show auth-web interface

dot1x eap

This command selects the transmit mode for the EAP packet. If the authentication feature is not enabled then EAP transmit mode is not enabled. The default setting discards EAP packets.

Syntax `dot1x eap {discard|forward|forward-untagged-vlan|forward-vlan}`

Parameter	Description
`discard`	Discard.
`forward`	Forward to all ports on the switch.
`forward-untagged-vlan`	Forward to ports with the same untagged VLAN.
`forward-vlan`	Forward to ports with the same VLAN.

Default The transmit mode is set to `discard` EAP packets by default.

Mode Global Configuration

Example To set the transmit mode of EAP packet to `forward` to forward EAP packets to all ports on the switch, use the following commands:

```
        awplus# configure terminal
awplus(config)# dot1x eap forward
```

To set the transmit mode of EAP packet to `discard` to discard EAP packets, use the following commands:

```
        awplus# configure terminal
awplus(config)# dot1x eap discard
```

To set the transmit mode of EAP packet to `forward-untagged-vlan` to forward EAP packets to ports with the same untagged vlan, use the following commands:

```
        awplus# configure terminal
awplus(config)# dot1x eap forward-untagged-vlan
```

To set the transmit mode of EAP packet to `forward-vlan` to forward EAP packets to ports with the same vlan, use the following commands:

```
        awplus# configure terminal
awplus(config)# dot1x eap forward-vlan
```

dot1x eapol-version

This command sets the EAPOL protocol version for EAP packets when 802.1X port authentication is applied.

Use the **no** variant of this command to set the EAPOL protocol version to 1.

The default EAPOL protocol version is version 1.

Syntax

```
dot1x eapol-version {1|2}
```

```
no dot1x eapol-version
```

Parameter	Description
1	EAPOL version.
2	EAPOL version.

Default The EAP version for 802.1X authentication is set to 1 by default.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

Example To set the EAPOL protocol version to 2 for port1.0.2, use the following commands:

```
           awplus# configure terminal
   awplus(config)# interface port1.0.2
awplus(config-if)# dot1x eapol-version 2
```

To set the EAPOL protocol version to the default version (1) for interface port1.0.2, use the following commands:

```
           awplus# configure terminal
   awplus(config)# interface port1.0.2
awplus(config-if)# no dot1x eapol-version
```

Validation Commands

show dot1x
show dot1x interface

dot1x initialize interface

This command initializes the 802.1X status on the specified interface, and attempts reauthentication.

Use this command to unauthorize a port, and attempt reauthentication on the specified interface, which may be a static channel (or static aggregator) or a dynamic (or LACP) channel group or a switch port.

Syntax `dot1x initialize interface <interface-list>`

Parameter	Description
<interface-list>	The interfaces or ports to configure. An interface-list can be: ■ an interface (e.g. `vlan2`), a switch port (e.g. `port1.0.12`), a static channel group (e.g. `sa3`) or a dynamic (LACP) channel group (e.g. `po4`) ■ a continuous range of interfaces, ports, static channel groups or dynamic (LACP) channel groups separated by a hyphen; e.g. `vlan2-8`, or `port1.0.1-1.0.24`, or `sa2-4`, or `po1-3` ■ a comma-separated list of the above; e.g. `port1.0.1,port1.0.8-1.0.24`. Do not mix interface types in a list The specified interfaces must exist.

Mode Privileged Exec

Example To initialize 802.1X port authentication on the interface `port1.0.2`, use the following command:

```
awplus# dot1x initialize interface port1.0.2
```

To unauthorize switch `port1.0.1` and attempt reauthentication on switch `port1.0.1` enter:

```
awplus# dot1x initialize interface port1.0.1
```

To unauthorize all switch ports for a 24 switch port device and attempt reauthentication enter:

```
awplus# dot1x initialize interface port1.0.1-port1.0.24
```

dot1x keytransmit

This command enables key transmission on the interface specified previously in Interface mode.

The **no** variant of this command disables key transmission on the interface specified.

Syntax

```
dot1x keytransmit

no dot1x keytransmit
```

Default Key transmission for port authentication is enabled by default.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

Usage Use this command to enable key transmission over an Extensible Authentication Protocol (EAP) packet between the authenticator and supplicant. Use the **no** variant of this command to disable key transmission.

Example To enable the key transmit feature on interface `port1.0.2`, after it has been disabled by negation, use the following commands:

```
         awplus# configure terminal
 awplus(config)# interface port1.0.2
awplus(config-if)# dot1x keytransmit
```

To disable the key transmit feature from the default startup configuration on interface `port1.0.2`, use the following commands:

```
         awplus# configure terminal
 awplus(config)# interface port1.0.2
awplus(config-if)# no dot1x keytransmit
```

Validation Commands

show dot1x
show dot1x interface

dot1x max-auth-fail

Use this command to configure the maximum number of login attempts for a supplicant (client device) using the **auth-fail vlan** feature, when using 802.1X port authentication on an interface.

The **no** variant of this command resets the maximum login attempts for a supplicant (client device) using the auth-fail vlan feature, to the default configuration of 3 login attempts.

Syntax

```
dot1x max-auth-fail <0-10>

no dot1x max-auth-fail
```

Parameter	Description
<0-10>	Specify the maximum number of login attempts for supplicants on an interface using 802.1X port authentication.

Default The default maximum number of login attempts for a supplicant on an interface using 802.1X port authentication is three (3) login attempts.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

Usage This command sets the maximum number of login attempts for supplicants on an interface. The supplicant is moved to the auth-fail VLAN from the Guest VLAN after the number of failed login attempts using 802.1X authentication is equal to the number set with this command.

See the related auth auth-fail vlan command on page 40.3. See also the section "Failed authentication VLAN" on page 39.12 for information about the auth-fail VLAN feature.

See the section "Limitations on allowed feature combinations" on page 39.13 for information about restrictions regarding combinations of authentication enhancements working together.

Examples To configure the maximum number of login attempts for a supplicant on interface `port1.0.2` to a single (1) login attempt, use the following commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.2
awplus(config-if)# dot1x max-auth-fail 1
```

To configure the maximum number of login attempts for a supplicant on interface `port1.0.2` to the default number of three (3) login attempts, use the following commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.2
awplus(config-if)# no dot1x max-auth-fail
```

Validation Commands show running-config

Related Commands auth auth-fail vlan
dot1x max-reauth-req
show dot1x interface

dot1x max-reauth-req

This command sets the number of reauthentication attempts before an interface is unauthorized.

The **no** variant of this command resets the reauthentication delay to the default.

Syntax

```
dot1x max-reauth-req <1-10>

no dot1x max-reauth-req
```

Parameter	Description
`<1-10>`	Specify the maximum number of reauthentication attempts for supplicants on an interface using 802.1X port authentication.

Default The default maximum reauthentication attempts for interfaces using 802.1X port authentication is two (2) reauthentication attempts, before an interface is unauthorized.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

Usage Use this command to set the maximum reauthentication attempts after failure.

Examples To configure the maximum number of reauthentication attempts for interface `port1.0.2` to a single (1) reauthentication request, use the following commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.2
awplus(config-if)# dot1x max-reauth-req 1
```

To configure the maximum number of reauthentication attempts for interface `port1.0.2` to the default maximum number of two (2) reauthentication attempts, use the following commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.2
awplus(config-if)# no dot1x max-reauth-req
```

Validation Commands show running-config

Related Commands dot1x max-auth-fail
show dot1x interface

dot1x port-control

This command enables 802.1X port authentication on the interface specified, and sets the control of the authentication port. When **port-control** is set to **auto**, the 802.1X authentication feature is executed on the interface, but only if the **aaa authentication dot1x** command has been issued.

The **no** variant of this command disables the port authentication on the interface specified.

Syntax

```
dot1x port-control {force-unauthorized|force-authorized|auto}
```

```
no dot1x port-control
```

Parameter	Description
`force-unauthorized`	Force port state to unauthorized. Specify to force a port to always be in an unauthorized state.
`force-authorized`	Force port state to authorized. Specify to force a port to always be in an authorized state.
`auto`	Allow port client to negotiate authentication. Specify to enable authentication on port.

Default 802.1X port control is disabled by default.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

Usage Use this command to force a port state. Note that all **dot1x** commands can only be applied to switch ports. They cannot be applied to dynamic (LACP) or static channel groups.

Example To enable port authentication on the interface `port1.0.2`, use the following commands:

```
          awplus#  configure terminal
  awplus(config)#  interface port1.0.2
awplus(config-if)# dot1x port-control auto
```

To enable port authentication force authorized on the interface `port1.0.2`, use the following commands:

```
          awplus#  configure terminal
  awplus(config)#  interface port1.0.2
awplus(config-if)# dot1x port-control force-authorized
```

To disable port authentication on the interface `port1.0.2`, use the following commands:

```
          awplus#  configure terminal
  awplus(config)#  interface port1.0.2
awplus(config-if)# no dot1x port-control
```

Validation Commands show dot1x interface

Related Commands aaa authentication dot1x

dot1x timeout tx-period

This command sets the transmit timeout for the authentication request on the specified interface.

The **no** variant of this command resets the transmit timeout period to the default (30 seconds).

Syntax

```
dot1x timeout tx-period <1-65535>

no dot1x timeout tx-period
```

Parameter	Description
<1-65535>	Seconds.

Default The default transmit period for port authentication is 30 seconds.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

Usage Use this command to set the interval between successive attempts to request an ID.

Example To set the transmit timeout period to 5 seconds on interface `port1.0.2`, use the below commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.2
awplus(config-if)# dot1x timeout tx-period 5
```

To reset transmit timeout period to the default (30 seconds) on interface `port1.0.2`, use the following commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.2
awplus(config-if)# no dot1x timeout tx-period
```

Validation Commands show dot1x
show dot1x interface

show debugging dot1x

Use this command to display the 802.1X debugging option set.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax `show debugging dot1x`

Mode User Exec and Privileged Exec

Usage This is a sample output from the `show debugging dot1x` command.

```
awplus# debug dot1x
```

```
awplus# show debugging dot1x
```

```
802.1X debugging status:
  802.1X events debugging is on
  802.1X timer debugging is on
  802.1X packets debugging is on
  802.1X NSM debugging is on
```

Example

```
awplus# show debugging dot1x
```

Related Commands debug dot1x

show dot1x

This command shows authentication information for dot1x (802.1X) port authentication.

If you specify the optional **all** parameter then this command also displays all authentication information for each port available on the switch.

For information on output options, see “Controlling “show” Command Output” on page 1.41.

Syntax `show dot1x [all]`

Parameter	Description
`all`	All.

Mode Privileged Exec

Example

```
awplus# show dot1x all
```

Table 38-1: Example output from the **show dot1x** command

```
awplus# show dot1x all
802.1X Port-Based Authentication Enabled
RADIUS server address: 150.87.18.89:1812
Next radius message id: 5
RADIUS client address: not configured
Authentication info for interface port1.0.12
portEnabled: true - portControl: Auto
portStatus: Authorized
reAuthenticate: disabled
reAuthPeriod: 3600
PAE: quietPeriod: 60 - maxReauthReq: 2 - txPeriod: 30
BE: suppTimeout: 30 - serverTimeout: 30
CD: adminControlledDirections: in
KT: keyTxEnabled: false
critical: disabled
guestVlan: disabled
dynamicVlanCreation: single-dynamic-vlan
    assignFailActionRule: deny
hostMode: multi-supplicant
    maxSupplicant: 1024
dot1x: enabled
    protocolVersion: 1
authMac: enabled
    method: PAP
    reauthRelearning: disabled
authWeb: enabled
    method: PAP
    lockCount: 3
    packetForwarding: disabled
```

Table 38-1: Example output from the **show dot1x** command (cont.)

```
supplicantMac: none
Supplicant name: manager
Supplicant address: 00d0.59ab.7037
    authenticationMethod: 802.1X Authentication
    portStatus: Authorized - currentId: 1
    abort:F fail:F start:F timeout:F success:T
    PAE: state: Authenticated - portMode: Auto
    PAE: reAuthCount: 0 - rxRespId: 0
    PAE: quietPeriod: 60 - maxReauthReq: 2 - txPeriod: 30
    BE: state: Idle - reqCount: 0 - idFromServer: 0
    CD: adminControlledDirections: in - operControlledDirections: in
    CD: bridgeDetected: false
    KR: rxKey: false
    KT: keyAvailable: false - keyTxEnabled: false
    criticalState: off
    dynamicVlanId: 2
802.1X statistics for interface port1.0.12
    EAPOL Frames Rx: 5 - EAPOL Frames Tx: 16
    EAPOL Start Frames Rx: 0 - EAPOL Logoff Frames Rx: 0
    EAP Rsp/Id Frames Rx: 3 - EAP Response Frames Rx: 2
    EAP Req/Id Frames Tx: 8 - EAP Request Frames Tx: 2
    Invalid EAPOL Frames Rx: 0 - EAP Length Error Frames Rx: 0
    EAPOL Last Frame Version Rx: 1 - EAPOL Last Frame Src:  00d0.59ab.7037
Authentication session statistics for interface port1.0.12
    session user name: manager
    session authentication method: Remote server
    session time: 19440 secs
    session terminate cause: Not terminated yet
Authentication Diagnostics for interface port1.0.12
    Supplicant address: 00d0.59ab.7037
    authEnterConnecting: 2
    authEaplogoffWhileConnecting: 1
    authEnterAuthenticating: 2
    authSuccessWhileAuthenticating: 1
    authTimeoutWhileAuthenticating: 1
    authFailWhileAuthenticating: 0
    authEapstartWhileAuthenticating: 0
    authEaplogoggWhileAuthenticating: 0
    authReauthsWhileAuthenticated: 0
    authEapstartWhileAuthenticated: 0
    authEaplogoffWhileAuthenticated: 0
    BackendResponses: 2
    BackendAccessChallenges: 1
    BackendOtherrequestToSupplicant: 3
    BackendAuthSuccess: 1
    BackendAuthFails: 0
```

show dot1x diagnostics

This command shows 802.1X authentication diagnostics for the specified interface (optional), which may be a static channel (or static aggregator) or a dynamic (or LACP) channel group or a switch port.

If no interface is specified then authentication diagnostics are shown for all interfaces.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax

```
show dot1x diagnostics [interface <interface-list>]
```

Parameter	Description
`interface`	Specify a port to show.
`<interface-list>`	The interfaces or ports to configure. An interface-list can be: ■ an interface (e.g. `vlan2`), a switch port (e.g. `port1.0.12`), a static channel group (e.g. `sa3`) or a dynamic (LACP) channel group (e.g. `po4`) ■ a continuous range of interfaces, ports, static channel groups or dynamic (LACP) channel groups separated by a hyphen; e.g. `vlan2-8`, or `port1.0.1-1.0.24`, or `sa2-4`, or `po1-3` ■ a comma-separated list of the above; e.g. `port1.0.1,port1.0.8-1.0.24`. Do not mix interface types in a list The specified interfaces must exist.

Mode Privileged Exec

Example See the sample output below showing 802.1X authentication diagnostics for `port1.0.12`:

```
awplus# show dot1x diagnostics interface port1.0.12
```

Output Figure 38-1: Example output from the **show dot1x diagnostics** command

```
Authentication Diagnostics for interface port1.0.12
    Supplicant address: 00d0.59ab.7037
       authEnterConnecting: 2
       authEaplogoffWhileConnecting: 1
       authEnterAuthenticating: 2
       authSuccessWhileAuthenticating: 1
       authTimeoutWhileAuthenticating: 1
       authFailWhileAuthenticating: 0
       authEapstartWhileAuthenticating: 0
       authEaplogoggWhileAuthenticating: 0
       authReauthsWhileAuthenticated: 0
       authEapstartWhileAuthenticated: 0
       authEaplogoffWhileAuthenticated: 0
       BackendResponses: 2
       BackendAccessChallenges: 1
       BackendOtherrequestToSupplicant: 3
       BackendAuthSuccess: 1
```

show dot1x interface

This command shows the status of 802.1X port-based authentication on the specified interface, which may be a static channel (or static aggregator) or a dynamic (or LACP) channel group or a switch port.

Use the optional **diagnostics** parameter to show authentication diagnostics for the specified interfaces. Use the optional **sessionstatistics** parameter to show authentication session statistics for the specified interfaces. Use the optional **statistics** parameter to show authentication diagnostics for the specified interfaces. Use the optional **supplicant** parameter to show the supplicant state for the specified interfaces.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax

```
show dot1x interface <interface-list>
   [diagnostics|sessionstatistics|statistics|supplicant [brief]]
```

Parameter	Description
<interface-list>	The interfaces or ports to configure. An interface-list can be: ■ an interface (e.g. `vlan2`), a switch port (e.g. `port1.0.12`), a static channel group (e.g. `sa3`) or a dynamic (LACP) channel group (e.g. `po4`) ■ a continuous range of interfaces, ports, static channel groups or dynamic (LACP) channel groups separated by a hyphen; e.g. `vlan2-8`, or `port1.0.1-1.0.24`, or `sa2-4`, or `po1-3` ■ a comma-separated list of the above; e.g. `port1.0.1,port1.0.8-1.0.24`. Do not mix interface types in a list The specified interfaces must exist.
`diagnostics`	Diagnostics.
`sessionstatistics`	Session Statistics.
`statistics`	Statistics.
`supplicant`	Supplicant.
`brief`	Brief summary of supplicant state.

Mode Privileged Exec

Example See the sample output below showing 802.1X authentication status for `port1.0.12`:

```
awplus# show dot1x interface port1.0.12
```

Table 38-2: Example output from the **show dot1x interface** command for a port

```
awplus#show dot1x interface port1.0.12
Authentication info for interface port1.0.12
   portEnabled: true - portControl: Auto
   portStatus: Authorized
   reAuthenticate: disabled
   reAuthPeriod: 3600
   PAE: quietPeriod: 60 - maxReauthReq: 2 - txPeriod: 30
   BE: suppTimeout: 30 - serverTimeout: 30
   CD: adminControlledDirections: in
   KT: keyTxEnabled: false
   critical: disabled
   guestVlan: disabled
   dynamicVlanCreation: single-dynamic-vlan
      assignFailActionRule: deny
   hostMode: multi-supplicant
      maxSupplicant: 1024
   dot1x: enabled
      protocolVersion: 1
   authMac: enabled
      method: PAP
      reauthRelearning: disabled
   authWeb: enabled
      method: PAP
      lockCount: 3
      packetForwarding: disabled
   supplicantMac: none
```

See the sample output below showing 802.1X authentication `sessionstatistics` for `port1.0.12`:

```
awplus# show dot1x interface port1.0.12 sessionstatistics
```

```
awplus#show dot1x interface port1.0.12 sessionstatistics
Authentication session statistics for interface port1.0.12
   session user name: manager
      session authentication method: Remote server
      session time: 19440 secs
      session terminat cause: Not terminated yet
```

See sample output below showing 802.1X authentication `diagnostics` for `port1.0.12`:

```
awplus# show dot1x interface port1.0.12 diagnostics
```

```
awplus#show dot1x interface port1.0.12 diagnostics
Authentication Diagnostics for interface port1.0.12
    Supplicant address: 00d0.59ab.7037
        authEnterConnecting: 2
        authEaplogoffWhileConnecting: 1
        authEnterAuthenticating: 2
        authSuccessWhileAuthenticating: 1
        authTimeoutWhileAuthenticating: 1
        authFailWhileAuthenticating: 0
        authEapstartWhileAuthenticating: 0
        authEaplogoggWhileAuthenticating: 0
        authReauthsWhileAuthenticated: 0
        authEapstartWhileAuthenticated: 0
        authEaplogoffWhileAuthenticated: 0
        BackendResponses: 2
        BackendAccessChallenges: 1
        BackendOtherrequestToSupplicant: 3
        BackendAuthSuccess: 1
```

See sample output below showing the `supplicant` on the interface `port1.0.12`:

```
awplus# show dot1x interface port1.0.12 supplicant
```

```
awplus#show dot1x interface port1.0.12 supplicant
authenticationMethod: dot1x
    totalSupplicantNum: 1
    authorizedSupplicantNum: 1
        macBasedAuthenticationSupplicantNum: 0
        dot1xAuthenticationSupplicantNum: 1
        webBasedAuthenticationSupplicantNum: 0
Supplicant name: manager
Supplicant address: 00d0.59ab.7037
    authenticationMethod: dot1x
    portStatus: Authorized - currentId: 4
    abort:F fail:F start:F timeout:F success:T
    PAE: state: Authenticated - portMode: Auto
    PAE: reAuthCount: 0 - rxRespId: 0
    PAE: quietPeriod: 60 - maxReauthReq: 2 - txPeriod: 30
    BE: state: Idle - reqCount: 0 - idFromServer: 3
    BE: suppTimeout: 30 - serverTimeout: 30
    CD: adminControlledDirections: in - operControlledDirections:
in
    CD: bridgeDetected: false
    KR: rxKey: false
    KT: keyAvailable: false - keyTxEnabled: false
```

See sample output below showing 802.1X (`dot1x`) authentication statistics for `port1.0.12`:

```
awplus# show dot1x statistics interface port1.0.12
```

```
awplus#show dot1x statistics interface port1.0.12
802.1X statistics for interface port1.0.12
    EAPOL Frames Rx: 5 - EAPOL Frames Tx: 16
    EAPOL Start Frames Rx: 0 - EAPOL Logoff Frames Rx: 0
    EAP Rsp/Id Frames Rx: 3 - EAP Response Frames Rx: 2
    EAP Req/Id Frames Tx: 8 - EAP Request Frames Tx: 2
    Invalid EAPOL Frames Rx: 0 - EAP Length Error Frames Rx: 0
    EAPOL Last Frame Version Rx: 1 - EAPOL Last Frame
Src:00d0.59ab.7037
```

Table 38-3: Parameters in the output of the **show dot1x interface** command

Parameter	Description
`portEnabled`	Interface operational status (Up-true/down-false).
`portControl`	Current control status of the port for 802.1X control.
`portStatus`	802.1X status of the port (authorized/unauthorized).
`reAuthenticate`	Reauthentication enabled/disabled status on port.
`reAuthPeriod`	Value holds meaning only if reauthentication is enabled.
`abort`	Indicates that authentication should be aborted when set to true.
`fail`	Indicates failed authentication attempt when set to false.
`start`	Indicates authentication should be started when set to true.
`timeout`	Indicates authentication attempt timed out when set to true.
`success`	Indicates authentication successful when set to true.
`state`	Current 802.1X operational state of interface.
`mode`	Configured 802.1X mode.
`reAuthCount`	Reauthentication count.
`quietperiod`	Time between reauthentication attempts.
`reAuthMax`	Maximum reauthentication attempts.
`BE`	Backend authentication state machine variables and constants.
`state`	State of the state machine.
`reqCount`	Count of requests sent to server.
`suppTimeout`	Supplicant timeout.
`serverTimeout`	Server timeout.
`maxReq`	Maximum requests to be sent.
`CD`	Controlled Directions State machine.
`adminControlledDirections`	Administrative value (Both/In).
`operControlledDirections`	Operational Value (Both/In).
`KR`	Key receive state machine.

Table 38-3: Parameters in the output of the **show dot1x interface** command (cont.)

Parameter	Description
`rxKey`	True when EAPOL-Key message is received by supplicant or authenticator. false when key is transmitted.
`KT`	Ket Transmit State machine.
`keyAvailable`	False when key has been transmitted by authenticator, true when new key is available for key exchange.
`keyTxEnabled`	Key transmission enabled/disabled status.

Related Commands

show auth-web diagnostics
show dot1x sessionstatistics
show dot1x statistics interface
show dot1x supplicant interface

show dot1x sessionstatistics

This command shows authentication session statistics for the specified interface, which may be a static channel (or static aggregator) or a dynamic (or LACP) channel group or a switch port.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax `show dot1x sessionstatistics [interface <interface-list>]`

Parameter	Description
`interface`	Specify a port to show.
`<interface-list>`	The interfaces or ports to configure. An interface-list can be: ■ an interface (e.g. `vlan2`), a switch port (e.g. `port1.0.12`), a static channel group (e.g. `sa3`) or a dynamic (LACP) channel group (e.g. `po4`) ■ a continuous range of interfaces, ports, static channel groups or dynamic (LACP) channel groups separated by a hyphen; e.g. `vlan2-8`, or `port1.0.1-1.0.24`, or `sa2-4`, or `po1-3` ■ a comma-separated list of the above; e.g. `port1.0.1,port1.0.8-1.0.24`. Do not mix interface types in a list The specified interfaces must exist.

Mode Privileged Exec

Example See sample output below showing 802.1X (`dot1x`) authentication session statistics for `port1.0.12`:

```
awplus# show dot1x sessionstatistics interface port1.0.12
```

```
Authentication session statistics for interface port1.0.12
    session user name: manager
        session authentication method: Remote server
        session time: 19440 secs
        session terminat cause: Not terminated yet
```

show dot1x statistics interface

This command shows the authentication statistics for the specified interface, which may be a static channel (or static aggregator) or a dynamic (or LACP) channel group or a switch port.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax `show dot1x statistics interface <interface-list>`

Parameter	Description
<interface-list>	The interfaces or ports to configure. An interface-list can be: ■ an interface (e.g. `vlan2`), a switch port (e.g. `port1.0.12`), a static channel group (e.g. `sa3`) or a dynamic (LACP) channel group (e.g. `po4`) ■ a continuous range of interfaces, ports, static channel groups or dynamic (LACP) channel groups separated by a hyphen; e.g. `vlan2-8`, or `port1.0.1-1.0.24`, or `sa2-4`, or `po1-3` ■ a comma-separated list of the above; e.g. `port1.0.1,port1.0.8-1.0.24`. Do not mix interface types in a list The specified interfaces must exist.

Mode Privileged Exec

Example See sample output below showing 802.1X authentication statistics for `port1.0.12`:

```
awplus# show dot1x statistics interface port1.0.12
```

```
802.1X statistics for interface port1.0.12
    EAPOL Frames Rx: 5 - EAPOL Frames Tx: 16
    EAPOL Start Frames Rx: 0 - EAPOL Logoff Frames Rx: 0
    EAP Rsp/Id Frames Rx: 3 - EAP Response Frames Rx: 2
    EAP Req/Id Frames Tx: 8 - EAP Request Frames Tx: 2
    Invalid EAPOL Frames Rx: 0 - EAP Length Error Frames Rx: 0
    EAPOL Last Frame Version Rx: 1 - EAPOL Last Frame
Src:00d0.59ab.7037
```

show dot1x supplicant

This command shows the supplicant state of the authentication mode set for the switch.

This command shows a summary when the optional **brief** parameter is used.

For information on output options, see “Controlling “show” Command Output” on page 1.41.

Syntax `show dot1x supplicant [<macadd>] [brief]`

Parameter	Description
`<macadd>`	MAC (hardware) address of the Supplicant.
`brief`	Brief summary of the Supplicant state.

Mode Privileged Exec

Example See sample output below showing the 802.1X authenticated supplicant on the switch:

```
awplus# show dot1x supplicant
```

```
authenticationMethod: dot1x
totalSupplicantNum: 1
authorizedSupplicantNum: 1
macBasedAuthenticationSupplicantNum: 0
dot1xAuthenticationSupplicantNum: 1
webBasedAuthenticationSupplicantNum: 0
Supplicant name: manager
Supplicant address: 00d0.59ab.7037
    authenticationMethod: dot1x
    portStatus: Authorized - currentId: 4
    abort:F fail:F start:F timeout:F success:T
    PAE: state: Authenticated - portMode: Auto
    PAE: reAuthCount: 0 - rxRespId: 0
    PAE: quietPeriod: 60 - maxReauthReq: 2 - txPeriod: 30
    BE: state: Idle - reqCount: 0 - idFromServer: 3
    BE: suppTimeout: 30 - serverTimeout: 30
    CD: adminControlledDirections: in - operControlledDirections:
in
    CD: bridgeDetected: false
    KR: rxKey: false
    KT: keyAvailable: false - keyTxEnabled: false
```

See sample output below showing the supplicant on the switch using the `brief` parameter:

```
awplus# show dot1x supplicant 00d0.59ab.7037 brief
```

```
Interface port1.0.12
   authenticationMethod: dot1x
   totalSupplicantNum: 1
   authorizedSupplicantNum: 1
       macBasedAuthenticationSupplicantNum: 0
       dot1xAuthenticationSupplicantNum: 1
       webBasedAuthenticationSupplicantNum: 0
Interface VID Mode MAC Address Status IP Address Username
========= === ==== ============ ======= =========== ========
port1.0.12 2 D 00d0.59ab.7037Authenticated 192.168.2.201 manager
```

Related Commands show dot1x supplicant interface

show dot1x supplicant interface

This command shows the supplicant state of the authentication mode set for the interface, which may be a static channel (or static aggregator) or a dynamic (or LACP) channel group or a switch port.

This command shows a summary when the optional **brief** parameter is used.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax `show dot1x supplicant interface <interface-list> [brief]`

Parameter	Description
<interface-list>	The interfaces or ports to configure. An interface-list can be: ■ an interface (e.g. `vlan2`), a switch port (e.g. `port1.0.12`), a static channel group (e.g. `sa3`) or a dynamic (LACP) channel group (e.g. `po4`) ■ a continuous range of interfaces, ports, static channel groups or dynamic (LACP) channel groups separated by a hyphen; e.g. `vlan2-8`, or `port1.0.1-1.0.24`, or `sa2-4`, or `po1-3` ■ a comma-separated list of the above; e.g. `port1.0.1,port1.0.8-1.0.24`. Do not mix interface types in a list The specified interfaces must exist.
`brief`	Brief summary of the Supplicant state.

Mode Privileged Exec

Example See sample output below showing the supplicant on the interface `port1.0.19`:

```
awplus# show dot1x supplicant interface port1.0.19
```

```
Interface port1.0.19
 authenticationMethod: dot1x
 totalSupplicantNum: 1
 authorizedSupplicantNum: 1
   macBasedAuthenticationSupplicantNum: 0
   dot1xAuthenticationSupplicantNum: 1
   webBasedAuthenticationSupplicantNum: 0
   otherAuthenticationSupplicantNum: 0

 Supplicant name: VCSPCVLAN10
 Supplicant address: 0000.cd07.7b60
   authenticationMethod: 802.1X
   portStatus: Authorized - currentId: 3
   abort:F fail:F start:F timeout:F success:T
   PAE: state: Authenticated - portMode: Auto
   PAE: reAuthCount: 0 - rxRespId: 0
   PAE: quietPeriod: 60 - maxReauthReq: 2
   BE: state: Idle - reqCount: 0 - idFromServer: 2
   CD: adminControlledDirections:in -
operControlledDirections:in
   CD: bridgeDetected: false
   KR: rxKey: false
   KT: keyAvailable: false - keyTxEnabled: false
```

See sample output below showing the supplicant on the switch using the `brief` parameter:

```
awplus# show dot1x supplicant interface brief
```

```
Interface port1.0.12
    authenticationMethod: dot1x
    totalSupplicantNum: 1
    authorizedSupplicantNum: 1
        macBasedAuthenticationSupplicantNum: 0
        dot1xAuthenticationSupplicantNum: 1
        webBasedAuthenticationSupplicantNum: 0
Interface VID Mode MAC Address    Status         IP Address     Username
========= === ==== ============ =======         ===========    ========
port1.0.12 2   D   00d0.59ab.7037Authenticated 192.168.2.201 manager
```

See the sample output below for static channel group (static aggregator) interface `sa1`:

```
awplus# show dot1x interface sa1 supplicant brief
```

```
awplus#show dot1x interface sa1 supplicant brief
Interface sa1
  authenticationMethod: dot1x
  totalSupplicantNum: 1
  authorizedSupplicantNum: 1
    macBasedAuthenticationSupplicantNum: 0
     dot1xAuthenticationSupplicantNum: 1
     webBasedAuthenticationSupplicantNum: 0
     otherAuthenticationSupplicantNum: 0

Interface   VID  Mode MAC Address    Status            IP Address      Username
=========== ==== ==== ============== ================= =============== ========
sa1         1    D    00d0.59ab.7037 Authenticated     --              test1
```

Related Commands show dot1x supplicant

undebug dot1x

This command applies the functionality of the **no debug dot1x** command on page 38.3.

Chapter 39: Authentication Introduction and Configuration

Authentication Introduction

Authentication commands enable you to specify three different types of device authentication: 802.1X authentication, MAC authentication, and Web-authentication. These are collectively called tri-authentication when applied to authenticate any devices connected to switch ports.

Tri-Authentication Introduction

The switch supports three types of authentication for devices that connect to switch ports:

- 802.1X authentication of devices connecting to switch ports
- MAC authentication of devices connecting to switch ports
- Web-authentication of devices connecting to switch ports

All three types can be configured to run simultaneously on a switch port. The simultaneous configuration and authentication of all three types on a port is called tri-authentication.

Tri-Authentication Configuration

Follow the below three steps to configure tri-authentication across a range of switch ports:

Step 1: Define the RADIUS Server:

Define the RADIUS Server where the switch will send authentication requests using the below commands:

```
        awplus# configure terminal

awplus(config)# radius-server host <ip-address> key
                <key-string>
```

These commands adds the RADIUS Server address and set parameters to the RADIUS server. The key parameter specifies the secret key for the server.

Note The RADIUS Server, where the switch sends authentication requests, could be the switch's own Local RADIUS Server. For information on how to configure Local RADIUS Server see Chapter 47, Local RADIUS Server Introduction and Configuration.

Step 2: Define the default authentication server lists:

Define the default authentication server lists for 802.1X authentication, Web-based authentication, and MAC-based authentication:

```
        awplus# configure terminal

awplus(config)# aaa authentication dot1x default group radius

awplus(config)# aaa authentication auth-web default group
                radius

awplus(config)# aaa authentication auth-mac default group
                radius
```

Step 3: Configure 802.1X, Web-based, and MAC-based authentication:

Configure 802.1X authentication, Web-authentication, MAC-authentication on switch ports to attach supplicant devices:

```
          awplus# configure terminal
  awplus(config)# interface <interface-range>
awplus(config-if)# switchport mode access
awplus(config-if)# switchport access vlan 1
awplus(config-if)# auth-web enable
awplus(config-if)# auth-mac enable
awplus(config-if)# dot1x port-control auto
awplus(config-if)# auth dynamic-vlan-creation
```

Configuring a Guest VLAN

You can configure 802.1X to accept a Dynamic VLAN assignment, or fall back to a Guest VLAN upon failure.

If 802.1X authentication has been configured on access ports in the network, you might still want to provide limited network access to those users whose devices do not have 802.1x supplicant enabled, or who have unrecognized authentication credentials.

The mechanism to achieve this is known as a guest VLAN. The idea is that if the users device fails 802.1X authentication, or is not even performing any 802.1X authentication, then its connection port can be put into the Guest VLAN.

To configure a switch to perform 802.1x authentication, and assign VLAN IDs to ports where devices authentication successfully, and put non-authenticated users into a Guest VLAN, proceed as follows:

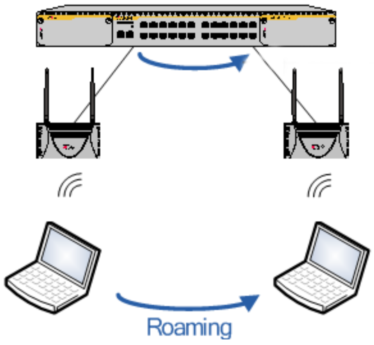
```
           awplus# configure terminal
   awplus(config)# radius-server host <ip-address> key
                   <key-string>
   awplus(config)# aaa authentication dot1x default group
                   radius
   awplus(config)# interface <interface-range>
awplus(config-if)# switchport mode access
awplus(config-if)# dot1x port-control auto
awplus(config-if)# auth dynamic-vlan-creation
awplus(config-if)# auth guest-vlan 100
```

Roaming Authentication

When network security is required, the usability of network security must be considered. The Roaming Authentication feature improves the usability of network security by enabling users to move within the network without requiring them to re-authenticate each time they move.

If a supplicant (client device) moves from one wireless access point to another wireless access point, and the wireless access points are connected to different ports, then the switch (authenticator) recognizes that the supplicant has been authenticated and accepts the supplicant without requiring re-authentication.

Figure 39-1: Diagram showing Roaming Authentication running on a standalone switch

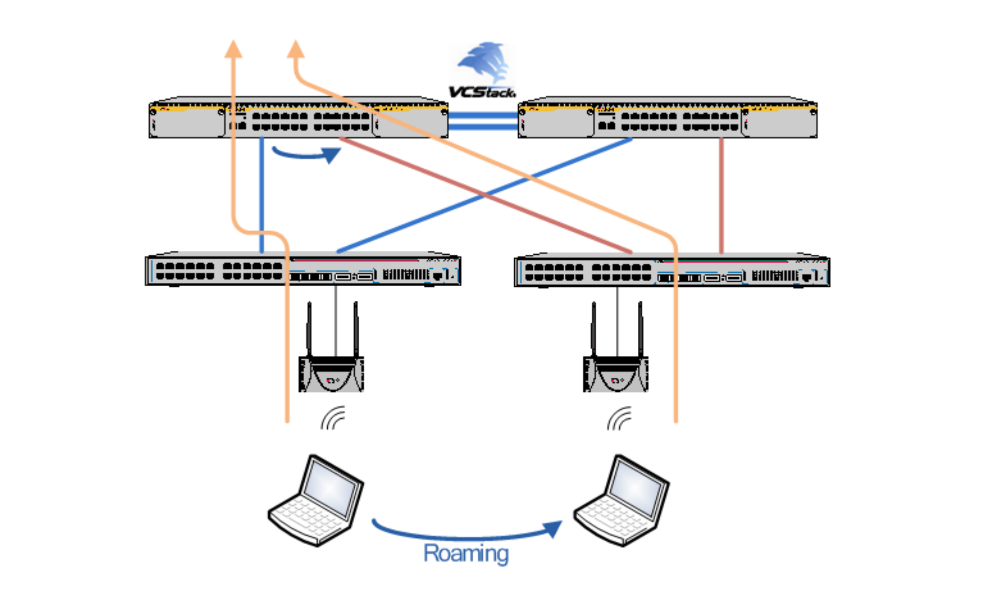


Web and MAC authentication are the authentication methods in a Wireless LAN environment, and 802.1X is the authentication method used for supplicants attached to edge switches.

Roaming Authentication is normally enabled using the auth roaming enable command on page 40.16 command. However, Roaming Authentication has been extended (with the auth roaming disconnected command on page 40.14) to work where an interface is link down. This allows you to enable supplicants to move from authenticated interfaces that are link down, without requiring re-authentication.

Roaming Authentication is available for use with the VCStack feature, and is available on static and dynamic (LACP) channel group interfaces.

Figure 39-2: Diagram showing Roaming Authentication running with VCStack



Roaming Authentication Overview

Without the Roaming Authentication feature enabled, if a supplicant moves from one switch port to another switch port, the supplicant's authenticated status, authentication, and assigned VLAN is deleted and the supplicant is re-authenticated so the supplicant can access the network, and all traffic from the supplicant is dropped while the supplicant is being re-authenticated.

With the Roaming Authentication feature enabled, a switch port inherits the status of a supplicant from the switch port that the supplicant was moved from. If the Roaming Authentication feature is enabled on a switch, then once a supplicant (client device) is authenticated on the switch it does not have to be re-authenticated if it moves between ports of that switch. Supplicant traffic is not dropped because there is no delay for re-authentication, during which the supplicant cannot access the network.

For example, when the Roaming Authentication feature is used in an wireless LAN environment with wireless access points, then the wireless clients can roam between wireless access points connected to different switch ports without re-authentication.

The Roaming Authentication feature also supports VCStack operation and works on defined static channel group (static aggregators) and dynamic channel group (LACP) interfaces. When VCStack and Roaming Authentication features are used together, the status of a supplicant is inherited from one aggregated interface to another aggregated interface over the stack.

See the auth roaming disconnected command on page 40.14 and the auth roaming enable command on page 40.16 for further information about configuring Roaming Authentication.

Roaming Authentication Feature Interactions

When the Roaming Authentication feature is disabled, a supplicant must be re-authenticated on the destination interface when it roams. When the Roaming Authentication is enabled, the following supplicant authentication status and information is inherited from the source interface:

- Authentication status
- Authentication method
- Supplicant MAC address
- Supplicant IP address
  (if an authenticated interface is configured for Web authentication)
- Supplicant name
- Authorized dynamic VLAN ID
- Authorized RADIUS server
- Reauthentication timer
  (if configured using the **auth timeout reauth-period** command on page 40.21)

Roaming Authentication is only supported between interfaces with the same authentication configuration. If source and destination interfaces have different authentication configuration then the supplicant will be re-authenticated at the destination interface.

When the host mode is set with the **auth host-mode** command on page 40.10, a supplicant is not authenticated on a destination interface, and the authentication status is deleted on the source interface.

When a supplicant moves from an interface with authentication configured to an interface without authentication configured, the supplicant's authentication status is deleted.

A supplicant is re-authenticated when it moves to a destination interface that is configured on a different VLAN than the VLAN that is configured for the source interface.

See the following Roaming Authentication feature interactions:

- Multiple Dynamic VLANs are supported when configured with the **auth dynamic-vlan-creation** command on page 40.6 using the **multi** parameter. Multiple Dynamic VLANs are disabled by default.
- Supplicants are re-authenticated on the destination interface if the VLAN ID changes when Single Dynamic VLANs are configured with the **auth dynamic-vlan-creation** command on page 40.6 the using the **single** parameter. Single Dynamic VLANs are disabled by default.
- The Roaming Authentication feature is supported on Guest VLANs configured by the **auth guest-vlan** command on page 40.8,

When the Roaming Authentication feature is configured for use on a stack with the VCStack feature, note that supplicants are initialized and re-authenticated if a VCStack failover occurs.

Unauthenticated Supplicant Traffic

When any authentication is configured on a switch port, the question arises as to what the switch does with packets that arrive into the switch port from unauthenticated supplicants.

Unauthenticated supplicants fall into three categories listed below:

- Newly attached supplicants, which are still in the process of their first authentication attempt
- Supplicants that have made an authentication attempt, but have failed authentication
- Supplicants that have been attached, but have not made an authentication attempt. For example, on a port that has only 802.1x authentication enabled, any supplicant that has no 802.1x client software will not be able to attempt 802.1x authentication.

In switches that are running the AlliedWare Plus^{TM} Operating System, packets from all these three categories of unathenticated supplicants are treated equally; no distinction is made between these three categories. The treatment of the traffic from unauthenticated supplicants does, however, depend on two factors:

- Whether a Guest VLAN has been configured on the switch port to which the supplicant is attached
- Whether Web authentication has been configured on the switch port to which the supplicant is attached

The rules governing the treatment of packets from unauthenticated supplicants are laid out in the table below:.

Table 39-1: Treatment of packets from unauthenticated supplicants

Switch port configuration	No Guest VLAN configured	Guest VLAN configured
Web authentication configured	Packets from unauthenticated supplicants are associated with the Native VLAN of the port. Packets from unauthenticated supplicants are processed according these rules: ■ Packets destined to the WebAuth server IP address/TCP port are forwarded to the server (which may well be the switch itself). ■ DHCP packets are sent to the CPU, to be processed by a local DHCP server, or relayed to another DHCP server, depending on the configuration of the switch. ■ DNS packets are forwarded to the CPU, and then sent on to a DNS server, if the switch is configured with a DNS server address. ■ ARP packets are forwarded to the CPU, and an ARP entry for the supplicant is learnt. ■ If web-auth forwarding is enabled for particular types of packets, then those packets will be forwarded within the Native VLAN ■ All other packets are dropped.	Packets from unauthenticated supplicants are associated with the Guest VLAN of the port. Packets from unauthenticated supplicants are processed according to these rules: ■ Packets destined to the WebAuth server IP address/TCP port are forwarded to the server (which may well be the switch itself). ■ DHCP packets are sent to the CPU, to be processed by a local DHCP server, or relayed to another DHCP server, depending on the configuration of the switch. ■ DNS packets are forwarded to the CPU, and then sent on to a DNS server, if the switch is configured with a DNS server address. ■ ARP packets are forwarded to the CPU, and an ARP entry for the supplicant is learnt. ■ Drop all other packets destined to the IP address of the Guest VLAN. ■ Layer 2 forward packets destined to other addresses within the Guest VLAN. ■ All other packets are dropped.

Table 39-1: Treatment of packets from unauthenticated supplicants

Switch port configuration	No Guest VLAN configured	Guest VLAN configured
No Web authentication configured	All non-eap packets from unauthenticated supplicants are dropped.	Packets from unauthenticated supplicants are associated with the Guest VLAN of the port. The packets are processed according to these rules: ■ Drop packets destined to the IP address of the Guest VLAN. ■ Layer 2 forward packets destined to other addresses within the Guest VLAN. ■ Drop all other packets.

Deciding when a supplicant fails authentication

Although the treatment of packets from unauthenticated supplicants does not differentiate between the three categories of supplicant, it is still useful to know for sure when the switch decides that a supplicant has failed authentication.

The rules for deciding that a supplicant has failed authentication are listed below for each type of authentication available:

Deciding when a supplicant fails 802.1X authentication

If the supplicant responds to EAP authentication requests, and the supplicant's authentication information is sent to the RADIUS server, and the RADIUS server replies with an Authentication-Reject, then the supplicant is immediately deemed to have failed authentication.

If the supplicant does not respond to EAP authentication requests, then the switch will resend the authentication requests up to a maximum number of attempts set by the command dot1x max-reauth-req (the default is 2). The interval between the attempts is set by the command dot1x timeout tx-period (the default is 30 seconds). If the supplicant still has not responded after this, it is deemed to have not attempted authentication.

See Chapter 38, 802.1X Commands for 802.1X authentication command information.

Deciding when a supplicant fails Web authentication

As soon as the supplicant attempts any web-browsing, the switch will intercept the web session, and present the supplicant with an authentication request page. If the user enters a username and password, and clicks the login button, then the switch will send the username and password to the RADIUS server. If the RADIUS server replies with an Authentication-Reject, then the supplicant is immediately deemed to have failed authentication.

Until the supplicant has attempted any web-browsing, or has received the authentication request page, but not yet clicked the login button, the supplicant is deemed to be not yet authenticated (as against not able to authenticate).

See Chapter 40, Authentication Commands for Web authentication command information.

Deciding when a supplicant fails MAC authentication

As soon as the supplicant sends any packet, the source MAC address from the packet will be sent to the RADIUS server for authentication. If the RADIUS server replies with an Authentication-Reject, then the supplicant is immediately deemed to have failed authentication.

With MAC auth there really is no concept of not-yet-attempted authentication, because authentication is attempted as soon as a supplicant sends a packet.

See Chapter 40, Authentication Commands for MAC authentication command information.

Authentication Enhancements

The authentication enhancements introduced in this release fall into three areas:

- Web-authentication Enhancements
  Improvements to Web-authentication
- Guest VLAN Enhancements
  Increased flexibility in the operation of the Guest VLAN
- Failed authentication VLAN
  Introduction of the auth-fail VLAN

See the section "Limitations on allowed feature combinations" on page 39.13 for information about restrictions regarding combinations of authentication enhancements working together.

Web-authentication Enhancements

Web-authentication can now operate as seamlessly as 802.1X authentication.

In previous releases there were limitations in the operation of Web-authentication. In particular, the authenticating switch had to be the default gateway for the client PC, or the user on the client PC had to browse explicitly to the IP address of the switch. As as result, previous web-authentication would not reliably interoperate with a client PC that had static IP configuration.

The aim of the enhancements in this release is to ensure that the client PC user is presented with the Web-authentication login page as soon as they start web browsing to **any** address, irrespective of the IP configuration (whether or not it is static or dynamic) on their client PC.

There are three aspects to the enhancements that have been implemented in order to do this:

- DHCP Server for Web-authentication
- Interception of clients' ARPs
- Proxy DNS response

DHCP Server for Web-authentication

In previous releases, a DHCP service could be configured on the authenticating switch, serving IP addresses in the subnet used on the Guest VLAN. While this did facilitate Web-authentication for client PCs with dynamic IP configuration, it was not an ideal solution, since the DHCP service was shared between Web-authentication clients and Guest VLAN users.

In this release there is now a DHCP server dedicated to serving IP addresses for use by Web-authentication clients.

See the auth-web-server dhcp ipaddress command on page 40.32 and the auth-web-server dhcp lease command on page 40.33 for details about configuring the Web-authentication DHCP Server.

Interception of clients' ARPs

A client PC's IP communications will always be preceded by sending out ARP (Address Resolution Protocol) requests for host addresses in its local subnet, or for its gateway address. If the IP address and gateway address have been statically configured on the client PC, and the subnet used in this static configuration is different to that on the authenticating switch, then the ARP requests will receive no reply, and the PC will not begin IP communications.

So, a switch operating as a Web Authenticator needs a method for replying to arbitrary ARP requests, to enable the client PC to proceed to the HTTP session required to perform Web-authentication.

The Web-authentication server can operate in three modes:

- **No interception**: only responds to ARP requests for its own IP address (this is the operation of the Web-authentication server in previous versions of AlliedWare Plus^{TM}).
- **Intercept mode**: will respond to ARP requests from any IP address that is in the same subnet as the switch's own IP address, and will provide its own MAC address in the ARP reply, irrespective of what IP address (within its own subnet) was being requested.
- **Promiscuous mode**: will respond to **any** ARP request, irrespective of whether the requested IP address is in the same subnet as the switch's IP address or not. It will provide its own MAC address in the ARP reply, irrespective of what IP address was being requested.

The addition of this functionality allows you to configure the Web-authentication server to interoperate with any static IP configuration on a client PC.

See the auth-web-server mode command on page 40.36 for command information about setting the Web-authentication mode.

Proxy DNS response

Typically, an HTTP session from a web browser is preceded by a DNS request for the IP address of the website the user wishes to browse to. If the DNS request does not receive a reply, then the web browser will never proceed to connect an HTTP session.

Hence the Web-authentication server needs a mechanism to reply to DNS requests, so that the Web-authentication session can begin. The Web-authentication modes also control the operation of Proxy DNS replies from the Web-authentication server as listed below:

- **No interception**: does not respond to DNS requests
- **Intercept mode**: responds to DNS request whose source IP address is within the same subnet as the IP address on the switch. The IP address provided as the resolution of the DNS lookup is the switch's own IP address, so that the subsequent HTTP traffic will be directed to the switch.
- **Promiscuous mode**: responds to DNS requests from any source IP address. The IP address provided as the resolution of the DNS lookup is the switch's own IP address, so that the subsequent HTTP traffic will be directed to the switch.

The combination of these enhancements ensure that, irrespective of the IP configuration on the client PC, Web-authentication will proceed smoothly.

See the auth-web-server mode command on page 40.36 for command information about setting the Web-authentication mode.

Guest VLAN Enhancements

In previous releases, traffic from unauthenticated supplicants in the Guest VLAN could only be L2 switched within the Guest VLAN.

As a result, it was not possible for DHCP requests from host in the Guest VLAN to be relayed to a DHCP server in another VLAN. Hence, for hosts in the Guest VLAN to obtain DHCP leases, the DHCP Server needed to have an interface in the Guest VLAN, or the authenticating switch needed to act as a DHCP Server. Either of these options could be inconvenient, or possibly even something of a security risk.

Additionally, supplicants in the Guest VLAN who needed to log into a Domain Controller, as part of becoming integrated into a NAC solution, could not access the Domain Controller if it was in another VLAN.

In this release, there is now an option to enable routing from the Guest VLAN. By default, traffic from unauthenticated supplicants in the Guest VLAN will still only be L2 switched within the Guest VLAN. But, if the **routing** parameter for the **auth guest vlan** command is configured, then the switch will route unauthenticated supplicants' traffic to other VLANs if required, and will relay their DHCP requests to servers in other VLANs if required.

See the **auth guest-vlan** command on page 40.8 for command information about Guest VLAN feature enhancements.

Failed authentication VLAN

The auth-fail VLAN feature allows the Network Administrator to separate the supplicants who attempted authentication, but failed, from the supplicants who did not attempt authentication.

This feature enables the Network Administrator to enact a security policy in which the supplicants who fail authentication are given extremely limited access, or are given access to remedial applications.

If the Guest VLAN and auth-fail VLAN are both configured on a switch, then a newly connected supplicant initially belongs to the Guest VLAN. If newly connected supplicants attempt 802.1X port authentication or Web-authentication and fail, then they are moved from the Guest VLAN to the auth-fail VLAN.

The criteria for how many failed authentication attempts are allowed before the supplicant is moved to the auth-fail VLAN differs, depending on the authentication method used.

If Web-authentication is used, then the supplicant is moved to the auth-fail VLAN after the first failed attempt. If 802.1X port authentication is used, then the supplicant is moved to the auth-fail VLAN after the number of failed attempts is equal to the value configured by the **dot1x max-auth-fail** command (by default, three failed 802.1X authentication attempts are allowed).

Note that the auth-fail VLAN feature is not applicable for MAC authentication. Supplicants failing MAC authentication remain in the Guest VLAN and will not move to the auth-fail VLAN.

See the **auth auth-fail vlan** command on page 40.3 and the **dot1x max-auth-fail** command on page 38.9 for command information about the failed authentication vlan feature when using 802.1X port authentication on an interface.

Limitations on allowed feature combinations

Note that the Web-authentication feature and enhancements cannot be used with the Guest VLAN or auth-fail VLAN features. For further limitation information see the below tables:

Table 39-2: Interoperation of authentication types with Guest VLAN and auth-fail VLAN

Authentication Type:	**Guest VLAN (without routing mode)**	**Guest VLAN (with routing mode)**	**Failed Authentication VLAN**
802.1X Port-based Authentication	No change in functionality from previous releases	Unauthorized supplicant can access Guest VLAN. Use ACL for security on the interface.	Failed authentication supplicant can access auth-fail VLAN. See limitations table below for ACL usage limitation.
MAC Authentication	No change in functionality from previous releases	Unauthorized supplicant can access Guest VLAN. Use ACL for security on the interface.	(Not Available)
Web-based Authentication (without intercept mode)	No change in functionality from previous releases	Unauthorized supplicant can access Guest VLAN. Use ACL for security on the interface.	Failed authentication supplicant can access auth-fail VLAN. See limitations table below for ACL usage limitation.
Web-based Authentication (with intercept mode)	(Not Available)	(Not Available)	(Not Available)

Table 39-3: Interactions between Guest VLAN and auth-fail VLAN

Authentication Feature:	**Guest VLAN (without routing mode)**	**Guest VLAN (with routing mode)**	**Failed Authentication VLAN**
Guest VLAN (without routing mode)	(Not Available)	(Not Available)	Cannot configure ACLs on the Guest VLAN when it is not in routing mode. The Guest VLAN without routing mode has reserved ACLs already attached to it.
Guest VLAN (with routing mode)	(Not Available)	(Not Available)	Configuration of ACLs for additional interface security is recommended.
Failed Authentication VLAN	Cannot configure ACLs on the Guest VLAN when it is not in routing mode. The Guest VLAN without routing mode has reserved ACLs already attached to it.	Configuration of ACLs for additional interface security is recommended.	(Not Available)

Chapter 40: Authentication Commands

Command List

This chapter provides an alphabetical reference for Authentication commands. For more information, see Chapter 39, Authentication Introduction and Configuration, and Chapter 42, AAA Commands.

auth auth-fail vlan

Use this command to enable the **auth-fail vlan** feature on the specified vlan interface. This feature assigns supplicants (client devices), which have failed port authentication, to the specified VLAN interface.

Use the **no** variant of this command to disable the **auth-fail vlan** feature for a specified VLAN interface.

Syntax

```
auth auth-fail vlan <1-4094>

no auth auth-fail vlan
```

Parameter	Description
<1-4094>	Assigns the VLAN ID to any supplicants that have failed port authentication.

Default The **auth-fail vlan** feature is disabled by default.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

Usage Use the **auth-fail vlan** feature when using Web-authentication instead of the Guest VLAN feature, when you need to separate networks where one supplicant (client device) requires authentication and another supplicant does not require authentication from the same interface.

This is because the DHCP lease time using the Web authentication feature is shorter, and the **auth fail vlan** feature enables assignment to a different VLAN if a supplicant fails authentication.

When using 802.1X port authentication, use a dot1x max-auth-fail command to set the maximum number of login attempts. Three login attempts are allowed by default for 802.1X port authentication before supplicants trying to authenticate are moved from the Guest VLAN to the auth-fail VLAN. See the "dot1x max-auth-fail" on page 38.9 for command information.

See the section "Failed authentication VLAN" on page 39.12 in Chapter 39, Authentication Introduction and Configuration for further overview information about the auth-fail VLAN feature, which allows the Network Administrator to separate the supplicants who attempted authentication, but failed, from the supplicants who did not attempt authentication.

See the section "Limitations on allowed feature combinations" on page 39.13 for information about restrictions regarding combinations of authentication enhancements working together.

Use appropriate ACLs (Access Control Lists) on interfaces for extra security if a supplicant allocated to the designated auth-fail vlan can access the same network as a supplicant on the Guest VLAN. For more information about ACL concepts, and configuring ACLs see Chapter 31, Access Control Lists Introduction. For more information about ACL commands see:

- Chapter 32, IPv4 Hardware Access Control List (ACL) Commands
- Chapter 33, IPv4 Software Access Control List (ACL) Commands
- Chapter 34, IPv6 Software Access Control List (ACL) Commands

Examples To enable **auth-fail vlan** for `port1.0.2` and assign VLAN 1`00,` use the following commands:

```
         awplus# configure terminal
 awplus(config)# interface port1.0.2
awplus(config-if)# auth auth-fail vlan 100
```

To disable the **auth-fail vlan** feature for `port1.0.2`, use the following commands:

```
         awplus# configure terminal
 awplus(config)# interface port1.0.2
awplus(config-if)# no auth auth-fail vlan
```

Validation Commands show running-config

Related Commands dot1x max-auth-fail
show dot1x
show dot1x interface

auth critical

This command enables the critical port feature on the interface. When the critical port feature is enabled on an interface, and all the RADIUS servers are unavailable, then the interface becomes authorized.

The **no** variant of this command disables critical port feature on the interface.

Syntax

```
auth critical

no auth critical
```

Default The critical port of port authentication is disabled.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

Examples To enable the critical port feature on interface `port1.0.2`, use the following commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.2
awplus(config-if)# auth critical
```

To disable the critical port feature on interface `port1.0.2`, use the following commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.2
awplus(config-if)# no auth critical
```

Validation Commands

show auth-web-server
show dot1x
show dot1x interface
show running-config

auth dynamic-vlan-creation

This command enables and disables the Dynamic VLAN assignment feature.

The Dynamic VLAN assignment feature allows a supplicant (client device) to be placed into a specific VLAN based on information returned from the RADIUS server during authentication, on a given interface.

Use the **no** variant of this command to disable the Dynamic VLAN assignment feature.

Syntax

```
auth dynamic-vlan-creation [rule {deny|permit}] [type {multi|single}]
```

```
no auth dynamic-vlan-creation
```

Parameter	Description
`rule`	VLAN assignment rule.
`deny`	Deny a differently assigned VLAN ID. This is the default rule.
`permit`	Permit a differently assigned VLAN ID.
`type`	Specifies whether multiple different VLANs can be assigned to supplicants (client devices) attached to the port, or whether only a single VLAN can be assigned to supplicants on the port.
`multi`	Multiple Dynamic VLAN.
`single`	Single Dynamic VLAN.

Default By default, the Dynamic VLAN assignment feature is disabled.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

Usage If the Dynamic VLAN assignment feature is enabled (disabled by default), VLAN assignment is dynamic. If the Dynamic VLAN assignment feature is disabled then RADUS attributes are ignored and configured VLANs are assigned to ports. Dynamic VLANs may be associated with authenticated MAC addresses if the **type** parameter is applied with the **rule** parameter.

The **rule** parameter deals with the case where there are multiple supplicants attached to a port, and the type parameter has been set to **single**-vlan. The parameter specifies how the switch should act if different VLAN IDs end up being assigned to different supplicants. The keyword value **deny** means that once a given VID has been assigned to the first supplicant, then if any subsequent supplicant is assigned a different VID, that supplicant is rejected. The keyword value **permit** means that once a given VID has been assigned to the first supplicant, then if any subsequent supplicant is assigned a different VID, that supplicant is accepted, but it is actually assigned the same VID as the first supplicant.

If you issue an **auth dynamic-vlan-creation** command without an optional **rule** parameter and a required **deny** or **permit** keyword value then a second supplicant with a different VLAN ID is rejected. It is not assigned to the first supplicant's VLAN. Issuing an a**uth dynamic-vlan-creation** command without an optional **rule** parameter has the same effect as issuing an **auth dynamic-vlan-creation rule deny** command rejecting supplicants with differing VIDs.

The **type** parameter specifies whether multiple different VLANs can be assigned to supplicants attached to the port, or whether only a single VLAN can be assigned to supplicants on the port. The **type** parameter can select the port base VLAN or the MAC base VLAN from the RADIUS VLAN ID. This can be used when the host-mode is set to multi-supplicant. For **single**-host ports, the VLAN ID will be assigned to the port. It is not supported with the Guest VLAN feature. Display the ID assigned using a **show vlan** command. For **multi**-host ports, the VLAN ID will be assigned to the MAC address of the authenticated supplicant. The VLAN ID assigned for the MAC Base VLAN is displayed using the **show platform table vlan** command.

Examples To enable the Dynamic VLAN assignment feature on interface `port1.0.2,` use the commands:

```
          awplus#  configure terminal
  awplus(config)#  interface port1.0.2
awplus(config-if)#  auth dynamic-vlan-creation
```

To disable the Dynamic VLAN assignment feature on interface `port1.0.2`, use the commands:

```
          awplus#  configure terminal
  awplus(config)#  interface port1.0.2
awplus(config-if)#  no auth dynamic-vlan-creation
```

Validation Commands

show dot1x
show dot1x interface
show running-config

Related Commands auth host-mode

auth guest-vlan

This command enables and configures the Guest VLAN feature on the interface specified by associating a Guest VLAN with an interface. This command does not start authentication. The supplicant's (client device's) traffic is associated with the native VLAN of the interface if its not already associated with another VLAN. The **routing** option enables routing from the Guest VLAN to another VLAN, so the switch can lease DHCP addresses and accept access to a limited network.

The **no** variant of this command disables the guest vlan feature on the interface specified.

Syntax

```
auth guest-vlan <1-4094> [routing]

no auth guest-vlan [routing]
```

Parameter	Description
<1-4094>	VLAN ID (VID).
`routing`	Enables routing from the Guest VLAN to other VLANs.

Default The Guest VLAN authentication feature is disabled by default.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

Usage The Guest VLAN feature may be used by supplicants (client devices) that have not attempted authentication, or have failed the authentication process. Note that if a port is in multi-supplicant mode with per-port dynamic VLAN configuration, after the first successful authentication, subsequent hosts cannot use the guest VLAN due to the change in VLAN ID. This may be avoided by using per-user dynamic VLAN assignment.

When using the Guest VLAN feature with the multi-host mode, a number of supplicants can communicate via a guest VLAN before authentication. A supplicant's traffic is associated with the native VLAN of the specified switch port. The supplicant must belong to a VLAN before traffic from the supplicant can be associated.

Note that you must first define the VLAN with the **vlan** command that you will assign as a guest VLAN using this command. Also note that 802.1X must first be enabled on the port.

Guest VLAN authentication cannot be enabled if DHCP snooping is enabled (service dhcp-snooping command on page 53.26), and vice versa.

The Guest VLAN feature in previous releases had some limitations that have been removed. Until this release the Guest VLAN feature could not lease the IP address to the supplicant using DHCP Server or DHCP Relay features unless Web authentication was also applied. When using NAP authentication, the supplicant should have been able to log on to a domain controller to gain certification, but the Guest VLAN would not accept access to another VLAN.

The Guest VLAN routing mode in this release overcomes these issues. With the Guest VLAN routing mode, the switch can lease DHCP addresses and accept access to a limited network.

See the section "Guest VLAN Enhancements" on page 39.11 for further overview information about the enhancements to the Guest VLAN feature.

See the section "Limitations on allowed feature combinations" on page 39.13 for information about restrictions regarding combinations of authentication enhancements working together.

Examples To define vlan100 and assign the guest VLAN feature to vlan100 on interface port1.0.2, and enable routing from the guest vlan to other VLANs, use the following commands:

```
          awplus# configure terminal
  awplus(config)# vlan database
awplus(config-vlan)# vlan 100
awplus(config-vlan)# exit
  awplus(config)# interface port1.0.2
awplus(config-if)# dot1x port-control auto
awplus(config-if)# auth guest-vlan 100 routing
```

To disable the guest vlan feature on interface port1.0.2, use the following commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.2
awplus(config-if)# no auth guest-vlan
```

Validation Commands

show dot1x
show dot1x interface
show running-config

Related Commands

dot1x port-control
vlan

auth host-mode

This command selects host mode on the interface. Multi-host is an extension to IEEE802.1X.

Use the **no** variant of this command to set host mode to the default setting (single host).

Syntax

```
auth host-mode {single-host|multi-host|multi-supplicant}
no auth host-mode
```

Parameter	Description
`single-host`	Single host mode.
`multi-host`	Multi host mode.
`multi-supplicant`	Multi supplicant (client device) mode.

Default The default host mode for port authentication is for a single host.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

Examples To set the host mode to multi-supplicant on interface `port1.0.2`, use the following commands:

```
           awplus# configure terminal
   awplus(config)# interface port1.0.2
awplus(config-if)# auth host-mode multi-supplicant
```

To set the host mode to default (single host) on interface `port1.0.2`, use the following commands:

```
           awplus# configure terminal
   awplus(config)# interface port1.0.2
awplus(config-if)# no auth host-mode
```

Validation Commands

show dot1x
show dot1x interface
show running-config

auth log

Use this command to configure the types of authentication feature log messages that are output to the log file.

Use the **no** variant of this command to remove either specified types or all types of authentication feature log messages that are output to the log file.

Syntax

```
auth log {dot1x|auth-mac|auth-web} {success|failure|logoff|all}
```

```
no auth log {do1x|auth-mac|auth-web} {success|failure|logoff|all}
```

Parameter	Description
`dot1x`	Specify only 802.1X authentication log messages are output to the log file.
`auth-mac`	Specify only MAC authentication log messages are output to the log file.
`auth-web`	Specify only Web authentication log messages are output to the log file.
`success`	Specify only successful authentication log messages are output to the log file.
`failure`	Specify only authentication failure log messages are output to the log file.
`logoff`	Specify only authentication logoff messages are output to the log file. Note that link down, age out and expired ping polling messages will be included.
`all`	Specify all types of authentication log messages are output to the log file Note that this is the default behavior for the authentication logging feature.

Default All types of authentication log messages are output to the log file by default.

Mode Interface Configuration

Examples To configure the logging of MAC authentication failures to the log file for supplicants (client devices) connected to interface `port1.0.2`, use the following commands:

```
           awplus# configure terminal
   awplus(config)# interface port1.0.2
awplus(config-if)# auth log auth-mac failure
```

To configure the logging of all types of authentication log messages to the log file for supplicants (client devices) connected to interface `port1.0.2`, use the following commands:

```
           awplus# configure terminal
   awplus(config)# interface port1.0.2
awplus(config-if)# no auth log all
```

Validation Commands show running-config

auth max-supplicant

This command sets the maximum number of supplicants (client devices) on the interface that can be authenticated. After this value is exceeded supplicants are not authenticated.

The **no** variant of this command resets the maximum supplicant number to the default (1024).

Syntax

```
auth max-supplicant <2-1024>
```

```
no auth max-supplicant
```

Parameter	Description
<2-1024>	Limit number.

Default The max supplicant of port authentication is 1024.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

Examples To set the maximum number of supplicants to `10` on interface `port1.0.2`, use the following commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.2
awplus(config-if)# auth max-supplicant 10
```

To reset the maximum number of supplicant to default on interface `port1.0.2`, use the following commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.2
awplus(config-if)# no auth max-supplicant
```

Validation Commands

show dot1x
show dot1x interface
show running-config

auth reauthentication

This command enables re-authentication on the interface specified in the Interface mode, which may be a static channel group (or static aggregator) or a dynamic (or LACP) channel group or a switch port.

Use the **no** variant of this command to disables reauthentication on the interface.

Syntax

```
auth reauthentication

no auth reauthentication
```

Default Reauthentication of port authentication is disabled by default.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

Examples To enable reauthentication on interface `port1.0.2`, use the following commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.2
awplus(config-if)# auth reauthentication
```

Validation Commands

show dot1x
show dot1x interface
show running-config

auth roaming disconnected

This command enables the roaming authentication feature on an authenticated interface that is link down. A supplicant (a client device) is not reauthenticated when moved between authenticated interfaces, providing both interfaces have the roaming authentication feature enabled before the supplicant is moved.

Use the auth roaming enable command before using this command. The auth roaming disconnected command on its own will have no effect on the operation of the switch. This command will only come into effect once the base Roaming Authentication feature is enabled, using the auth roaming enable command.

The **no** variant of this command disables the roaming authentication feature on an interface, and forces a supplicant to be reauthenticated when moving between interfaces.

See "Roaming Authentication" on page 39.4 for further information about this feature.

Syntax

```
auth roaming disconnected

no auth roaming disconnected
```

Default The roaming authentication `disconnected` feature is disabled by default on an interface. Authentication status for a roaming supplicant is deleted by default when an interface goes down.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

Usage This command allows a supplicant to move to another authenticating interface without reauthentication, if the link is down for the interface that the supplicant is moved from.

Note that 802.1X port authentication, or MAC authentication, or Web Authentication must first be enabled on an interface to use this feature. The port that the supplicant is moving to must have the same authentication configuration as the port the supplicant is moving from.

Configure auth roaming enable on an interface before configuring auth roaming disconnected if you require auth roaming disconnected configured on an interface for a roaming supplicant.

Roaming authentication cannot be enabled if DHCP snooping is enabled (service dhcp-snooping command on page 53.26), and vice versa.

Examples To enable roaming authentication `disconnected` feature for `port1.0.2`, after enabling 802.1x authentication and enabling roaming authentication `enable`, use the commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.2
awplus(config-if)# dot1x port-control auto
awplus(config-if)# auth roaming enable
awplus(config-if)# auth roaming disconnected
```

To disable roaming authentication disconnected feature for `port1.0.2`, use the commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.2
awplus(config-if)# no auth roaming disconnected
```

Validation Commands

show running-config

Related Commands

auth-mac enable
auth roaming enable
auth-web enable
dot1x port-control
show auth-mac interface
show auth-web interface
show dot1x interface

auth roaming enable

This command enables the roaming authentication feature on an authenticated interface that is link up. A supplicant (a client device) is not reauthenticated when moved between authenticated interfaces, providing both interfaces have the roaming authentication feature enabled before the supplicant is moved.

Use the auth roaming enable command before using auth roaming disconnected command. The auth roaming disconnected command on its own will have no effect on the operation of the switch. This command will only come into effect once the base Roaming Authentication feature is enabled, using the auth roaming enable command.

The **no** variant of this command disables the roaming authentication feature on an interface, and forces a supplicant to be reauthenticated when moving between interfaces.

See "Roaming Authentication" on page 39.4 for further information about this feature.

Syntax

```
auth roaming enable
```

```
no auth roaming enable
```

Default The roaming authentication enable feature is disabled by default on an interface. Authentication status for a roaming supplicant is deleted by default when an interface goes down.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

Usage This command allows a supplicant to move to another authenticating interface without reauthentication, providing the link is up for the interface that the supplicant is moved from.

Note that 802.1X port authentication, or MAC authentication, or Web Authentication must first be enabled on an interface to use this feature. The port that the supplicant is moving to must have the same authentication configuration as the port the supplicant is moving from.

Configure auth roaming enable on an interface before configuring auth roaming disconnected if you require auth roaming disconnected configured on an interface for a roaming supplicant.

Roaming authentication cannot be enabled if DHCP snooping is enabled (service dhcp-snooping command on page 53.26), and vice versa.

Examples To enable the roaming authentication enable feature for interface `port1.0.4`, after enabling 802.1x authentication, since an authentication method is required, use the following commands:

```
           awplus#  configure terminal
   awplus(config)#  interface port1.0.4
awplus(config-if)#  dot1x port-control auto
awplus(config-if)#  auth roaming enable
```

To disable roaming authentication enable for `port1.0.4`, use the following commands:

```
           awplus#  configure terminal
   awplus(config)#  interface port1.0.4
awplus(config-if)#  no auth roaming enable
```

Validation Commands	show running-config
Related Commands	auth-mac enable auth roaming disconnected auth-web enable dot1x port-control show auth-mac interface show auth-web interface show dot1x interface

auth supplicant-mac

This command adds a supplicant (client device) mac address on a given interface with the parameters as specified in the table below.

Use the **no** variant of this command to delete the supplicant MAC address added by the **auth supplicant-mac** command, and resets to the default for the supplicant parameter.

Syntax

```
auth supplicant <mac-addr>
    [max-reauth-req <1-10>]
    [port-control {auto | force-authorized | force-unauthorized}]
    [quiet-period <1-65535>]
    [reauth-period <1-4294967295>]
    [supp-timeout <1-65535>]
    [server-timeout <1-65535>][reauthentication]

no auth supplicant-mac <macadd> [reauthentication]
```

Parameter	Description
<mac-addr>	MAC (hardware) address of the Supplicant entry in HHHH.HHHH.HHHH MAC address hexadecimal format.
port-control	Port control commands.
auto	Allow port client to negotiate authentication.
force-authorized	Force port state to authorized.
force-unauthorized	Force port state to unauthorized.
quiet-period	Quiet period in the HELD state (default 60 seconds).
<1-65535>	Seconds for quiet period.
reauth-period	Seconds between reauthorization attempts (default 3600 seconds).
<1-4294967295>	Seconds for reauthorization attempts (reauth-period).
supp-timeout	Supplicant response timeout (default 30 seconds).
<1-65535>	Seconds for supplicant response timeout.
server-timeout	Authentication server response timeout (default 30 seconds).
<1-65535>	Seconds for authentication server response timeout.
reauthentication	Enable reauthentication on a port.
max-reauth-req	No of reauthentication attempts before becoming unauthorized (default 2).
<1-10>	Count of reauthentication attempts.

Default No supplicant MAC address for port authentication exists by default until first created with the **auth supplicant-mac** command. The defaults for parameters applied are as shown in the table.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

Examples To add the supplicant MAC address 0009.41A4.5943 to force authorized port control for interface port1.0.2, use the following commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.2
awplus(config-if)# auth supplicant-mac 0009.41A4.5943 port-
                   control force-authorized
```

To delete the supplicant MAC address 0009.41A4.5943 for interface port1.0.2, use the following commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.2
awplus(config-if)# no auth supplicant-mac 0009.41A4.5943
```

To reset reauthentication to disable for the supplicant MAC address 0009.41A4.5943, for interface port1.0.2 use the following commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.2
awplus(config-if)# no auth supplicant-mac 0009.41A4.5943
                   reauthentication
```

Validation Commands

show dot1x
show dot1x interface
show running-config

auth timeout quiet-period

This command sets the time period for which the authentication request is not accepted on a given interface, after the authentication request has failed an authentication.

Use the **no** variant of this command to reset quiet period to the default (60 seconds).

Syntax

```
auth timeout quiet-period <1-65535>
no auth timeout quiet-period
```

Parameter	Description
<1-65535>	Seconds.

Default The quiet period of port authentication is 60 seconds.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

Examples To set the quiet period to `10` for interface `port1.0.2`, use the following commands:

```
        awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# auth timeout quiet-period 10
```

To reset the quiet period to the default (60 seconds) for interface `port1.0.2`, use the following commands:

```
        awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# no auth timeout quiet-period
```

auth timeout reauth-period

This command sets the timer for reauthentication on a given interface. The re-authentication for the supplicant (client device) is executed at this timeout. The timeout is only applied if the **auth reauthentication** command is applied.

Use the **no** variant of this command to reset the **reauth-period** parameter to the default (3600 seconds).

Syntax

```
auth timeout reauth-period <1-4294967295>
```

```
no auth timeout reauth-period
```

Parameter	Description
<1-4294967295>	Seconds.

Default The default reauthentication period for port authentication is 3600 seconds, when reauthentication is enabled on the port.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

Examples To set the reauthentication period to 1 day for interface `port1.0.2`, use the following commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.2
awplus(config-if)# auth timeout reauth-period 86400
```

To reset the reauthentication period to the default (3600 seconds) for interface `port1.0.2`, use the following commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.2
awplus(config-if)# no auth timeout reauth-period
```

Validation Commands

show dot1x
show dot1x interface

show running-config

Related Commands auth reauthentication

auth timeout server-timeout

This command sets the timeout for the waiting response from the RADIUS server on a given interface.

The **no** variant of this command resets the server-timeout to the default (30 seconds).

Syntax

```
auth timeout server-timeout <1-65535>

no auth timeout server-timeout
```

Parameter	Description
<1-65535>	Seconds.

Default The server timeout for port authentication is 30 seconds.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

Examples To set the server timeout to 120 seconds for interface `port1.0.2`, use the following commands:

```
         awplus#  configure terminal
  awplus(config)#  interface port1.0.2
awplus(config-if)#  auth timeout server-timeout 120
```

To set the server timeout to the default (30 seconds) for interface `port1.0.2`, use the following commands:

```
         awplus#  configure terminal
  awplus(config)#  interface port1.0.2
awplus(config-if)#  no auth timeout server-timeout
```

Validation Commands

show dot1x
show dot1x interface
show running-config

auth timeout supp-timeout

This command sets the timeout of the waiting response from the supplicant (client device) on a given interface.

The **no** variant of this command resets the supplicant timeout to the default (30 seconds).

Syntax

```
auth timeout supp-timeout <1-65535>
```

```
no auth timeout supp-timeout
```

Parameter	Description
<1-65535>	Seconds.

Default The supplicant timeout of port authentication is 30 seconds.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

Examples To set the server timeout to 2 seconds for interface `port1.0.2`, use the following commands:

```
          awplus#  configure terminal
  awplus(config)#  interface port1.0.2
awplus(config-if)# auth timeout supp-timeout 2
```

To reset the server timeout to the default (30 seconds) for interface `port1.0.2`, use the following commands:

```
          awplus#  configure terminal
  awplus(config)#  interface port1.0.2
awplus(config-if)# no auth timeout supp-timeout
```

Validation Commands

show dot1x
show dot1x interface
show running-config

auth-mac enable

This command enables MAC based authentication on the interface specified in the Interface command mode.

Use the **no** variant of this command to disable MAC based authentication on an interface.

Syntax

```
auth-mac enable
no auth-mac enable
```

Default MAC authentication is disabled by default.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

Usage Enabling **spanning-tree edgeport** on ports after enabling MAC based authentication avoids unnecessary re-authentication when the port state changes, which does not happen when spanning tree edgeport is enabled. Note that re-authentication is correct behavior without **spanning-tree edgeport** enabled.

Applying **switchport mode access** on ports is also good practice to set the ports to access mode with ingress filtering turned on, whenever ports for MAC authentication are in a VLAN.

Examples To enable MAC authentication on interface `port1.0.2` and enable spanning tree edgeport to avoid unnecessary re-authentication, use the following commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.2
awplus(config-if)# auth-mac enable
awplus(config-if)# spanning-tree edgeport
awplus(config-if)# switchport mode access
```

To disable MAC authentication on interface `port1.0.2`, use the following commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.2
awplus(config-if)# no auth-mac enable
```

Validation Commands

show auth-mac
show auth-mac interface
show running-config

Related Commands

aaa accounting auth-mac default
aaa authentication auth-mac
spanning-tree edgeport (RSTP and MSTP)
switchport mode access

auth-mac method

This command sets the type of authentication method for MAC authentication that is used with RADIUS on the interface specified in the Interface command mode.

The **no** variant of this command resets the authentication method used to the default method (PAP) as the RADIUS authentication method used by the MAC authentication.

Syntax

```
auth-mac method [eap-md5|pap]
```

```
no auth-mac method
```

Parameter	Description
`eap-md5`	Enable EAP-MD5 of authentication method.
`pap`	Enable PAP of authentication method.

Default The mac authentication method is PAP.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

Examples To set the MAC authentication method to `pap` on interface `port1.0.2`, use the following commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.2
awplus(config-if)# auth-mac method pap
```

To set the MAC authentication method to the default on interface `port1.0.2`, use the following commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.2
awplus(config-if)# no auth-mac method
```

Validation Commands

show auth-mac
show auth-mac interface
show running-config

auth-mac reauth-relearning

This command sets the MAC address learning of the supplicant (client device) to re-learning for re-authentication on the interface specified in the Interface command mode.

Use the **no** variant of this command to disable the auth-mac re-learning option.

Syntax

```
auth-mac reauth-relearning

no auth-mac reauth-relearning
```

Default Re-learning for port authentication is disabled by default.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

Examples To enable the re-authentication re-learning feature on interface `port1.0.2`, use the following commands:

```
         awplus#  configure terminal
  awplus(config)#  interface port1.0.2
awplus(config-if)#  auth-mac reauth-relearning
```

To disable the re-authentication re-learning feature on interface `port1.0.2`, use the following commands:

```
         awplus#  configure terminal
  awplus(config)#  interface port1.0.2
awplus(config-if)#  no auth-mac reauth-relearning
```

Validation Commands

show auth-mac
show auth-mac interface
show running-config

auth-web enable

This command enables Web-based authentication in Interface mode on the interface specified.

Use the **no** variant of this command to disable Web-based authentication on an interface.

Syntax

```
auth-web enable
no auth-web enable
```

Default Web authentication is disabled by default.

Mode Interface Configuration for a static channel or a switch port.

Usage Web-based authentication cannot be enabled if DHCP snooping is enabled (service dhcp-snooping command on page 53.26), and vice versa.

Examples To enable Web authentication on static-channel-group 2, use the following commands:

```
         awplus# configure terminal
   awplus(config)# interface port1.0.2
awplus(config-if)# static-channel-group 2
awplus(config-if)# exit
   awplus(config)# interface sa2
awplus(config-if)# auth-web enable
```

To disable Web authentication on static-channel-group 2, use the following commands:

```
         awplus# configure terminal
   awplus(config)# interface port1.0.2
awplus(config-if)# static-channel-group 2
awplus(config-if)# exit
   awplus(config)# interface sa2
awplus(config-if)# no auth-web enable
```

Validation Commands

show auth-web
show auth-web interface
show running-config

Related Commands

aaa accounting auth-web default
aaa authentication auth-web

auth-web forward

This command enables the web authentication packet forwarding feature on the interface specified. This command also enables ARP forwarding, and adds forwarded packets to the **tcp** or **udp** port number specified.

The **no** variant of this command disables or deletes the packet forwarding feature on the interface.

Syntax

```
auth-web forward {arp|dhcp|dns|tcp <1-65535>|udp <1-65535>}
no auth-web forward [arp|dhcp|dns|tcp <1-65535>|udp <1-65535>]
```

Parameter	Description
`arp`	Enable forwarding of ARP.
`dhcp`	Enable forwarding of DHCP (67/udp).
`dns`	Enable forwarding of DNS (53/udp).
`tcp`	Enable forwarding of TCP specified port number.
<1-65535>	TCP Port number.
`udp`	Enable forwarding of UDP specified port number.
<1-65535>	UDP Port number.

Default Packet forwarding for port authentication is disabled by default.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

Examples To enable the arp forwarding feature on interface `port1.0.2`, use the following commands:

```
        awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# auth-web forward arp
```

To add the tcp forwarding port 137 on interface `port1.0.2`, use the following commands:

```
        awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# auth-web forward tcp 137
```

To disable the ARP forwarding feature on interface `port1.0.2`, use the following commands:

```
        awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# no auth-web forward arp
```

To delete the tcp forwarding port 137 on interface port1.0.2, use the following commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.2
awplus(config-if)# no auth-web forward tcp 137
```

To delete the all of tcp forwarding on interface port1.0.2, use the following commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.2
awplus(config-if)# no auth-web forward tcp
```

Validation Commands

show auth-web
show auth-web interface
show running-config

auth-web max-auth-fail

This command sets the number of authentication failures allowed before rejecting further authentication requests. When the supplicant (client device) fails more than has been set to the maximum number of authentication failures then login requests are refused during the quiet period.

The **no** variant of this command resets the maximum number of authentication failures to the default (3 authentication failures).

Syntax

```
auth-web max-auth-fail <0-10>

no auth-web max-auth-fail
```

Parameter	Description
<0-10>	Lock count specified.

Default The **max-auth-fail** lock counter is set to 3 authentication failures by default.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

Examples To set the lock count to 5 on interface port1.0.2, use the following commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.2
awplus(config-if)# auth-web max-auth-fail 5
```

To set the lock count to the default on interface port1.0.2, use the following commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.2
awplus(config-if)# no auth-web max-auth-fail
```

Validation Commands show auth-web
show auth-web interface
show running-config

Related Commands auth timeout quiet-period

auth-web method

This command sets the authentication method of WEB authentication that is used with RADIUS on the interface specified.

The **no** variant of this command sets the authentication method to PAP for the interface specified when Web authentication is also used with the RADIUS authentication method.

Syntax

```
auth-web method {eap-md5|pap}
```

```
no auth-web method
```

Parameter	Description
`eap-md5`	Enable EAP-MD5 as the authentication method.
`pap`	Enable PAP as the authentication method.

Default The web authentication method is set to PAP by default.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

Example To set the web authentication method to eap-md5 on interface `port1.0.2`, use the following commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.2
awplus(config-if)# auth-web method eap-md5
```

Validation Commands

show auth-web
show auth-web interface
show running-config

auth-web-server dhcp ipaddress

Use this command to assign an IP address and enable the DHCP service on the web authentication server for supplicants (client devices).

Use the **no** variant of this command to remove an IP address and disable the DHCP service on the web authentication server for supplicants.

Syntax

```
auth-web-server dhcp ipaddress <ip-address/prefix-length>

no auth-web-server dhcp ipaddress
```

Parameter	Description
<ip-addr/ prefix-length>	The IPv4 address and prefix length assigned for the DHCP service on the web authentication server for supplicants.

Default No IP address for the web authentication server is set by default.

Mode Global Configuration

Usage See the section "DHCP Server for Web-authentication" on page 39.10 in Chapter 39, Authentication Introduction and Configuration for further overview information about the Web-authentication enhancements, allowing Web-authentication to work as seamlessly as 802.1X authentication.

See the section "Limitations on allowed feature combinations" on page 39.13 for information about restrictions regarding combinations of authentication enhancements working together.

Examples To assign the IP address `10.0.0.1` to the web authentication server, use the following commands:

```
        awplus# configure terminal
awplus(config)# auth-web-server dhcp ip address 10.0.0.1/8
```

To remove an IP address on the web authentication server, use the following commands:

```
        awplus# configure terminal
awplus(config)# no auth-web-server dhcp ip address
```

Validation Commands show running-config

Related Commands show auth-web-server
auth-web-server dhcp lease

auth-web-server dhcp lease

Use this command to set the DHCP lease time for supplicants (client devices) using the DHCP service on the web authentication server.

Use the **no** variant of this command to reset to the default DHCP lease time for supplicants using the DHCP service on the web authentication server.

Syntax

```
auth-web-server dhcp lease <20-60>

no auth-web-server dhcp lease
```

Parameter	Description
<20-60>	DHCP lease time for supplicants using the DHCP service on the web authentication server in seconds.

Default The default DHCP lease time for supplicants using the DHCP service on the web authentication server is set to 30 seconds.

Mode Global Configuration

Usage See the section "DHCP Server for Web-authentication" on page 39.10 in Chapter 39, Authentication Introduction and Configuration for further overview information about the Web-authentication enhancements, allowing Web-authentication to work as seamlessly as 802.1X authentication.

See the section "Limitations on allowed feature combinations" on page 39.13 for information about restrictions regarding combinations of authentication enhancements working together.

Examples To set the DHCP lease time to 1 minute for supplicants using the DHCP service on the web authentication server, use the following commands:

```
        awplus# configure terminal
awplus(config)# auth-web-server dhcp lease 60
```

To reset the DHCP lease time to the default setting (30 seconds) for supplicants using the DHCP service on the web authentication server, use the following commands:

```
        awplus# configure terminal
awplus(config)# no auth-web-server dhcp lease
```

Validation Commands show running-config

Related Commands show auth-web-server
auth-web-server dhcp ipaddress

auth-web-server http-redirect

This command enables the HTTP redirect feature on every interface on which web-based port authentication is enabled. When the HTTP redirect feature is enabled, any HTTP request received on an unauthorized interface is redirected to the web authentication server automatically.

Use the **no** variant of this command to disable the HTTP redirect feature.

Syntax

```
auth-web-server http-redirect
no auth-web-server http-redirect
```

Default The HTTP redirect feature is enabled by default.

Mode Global Configuration

Examples To disable the HTTP redirect feature, use the following commands:

```
         awplus# configure terminal
 awplus(config)# no auth-web-server http-redirect
```

To re-enable the HTTP redirect feature, use the following commands:

```
         awplus# configure terminal
 awplus(config)# auth-web-server http-redirect
```

Validation Commands

show auth-web
show auth-web-server
show running-config

auth-web-server ipaddress

This command sets the IP address for the web authentication server.

Use the **no** variant of this command to delete the IP address for the web authentication server.

Syntax

```
auth-web-server ipaddress <ip-address>

no auth-web-server ipaddress
```

Parameter	Description
<ip-address>	Web authentication server dotted decimal IP address in A.B.C.D format.

Default The web authentication server address on the system is not set by default.

Mode Global Configuration

Examples To set the IP address 10.0.0.1 to the web authentication server, use the following commands:

```
        awplus# configure terminal
awplus(config)# auth-web-server ipaddress 10.0.0.1
```

To delete the IP address from the web authentication server, use the following commands:

```
        awplus# configure terminal
awplus(config)# no auth-web-server ipaddress
```

Validation Commands

show auth-web
show auth-web-server
show running-config

auth-web-server mode

Use this command with required keyword to configure an intercept mode (from the intercept, none, or promiscuous modes available) on the web authentication server for supplicants (client devices). The intercept modes available affect the interception of clients' ARPs and the proxy DNS response when using Web-authentication. These enhancements ensure that Web-authentication will proceed smoothly irrespective of the IP configuration on the client PC.

Use the **no** variant of this command to disable the intercept mode (either the intercept, none, or promiscuous intercept modes) configured on the web authentication server for supplicants.

Syntax

```
auth-web-server mode {intercept|none|promiscuous}

no auth-web-server mode {intercept|promiscuous}
```

Parameter	Description
`intercept`	Selecting this parameter results in web authentication server on the switch intercepting and replying to ARP and DNS messages from the same interface and IP address.
`none`	Selecting this parameter disables the intercept mode on the web authentication server. No ARP and DNS messages are intercepted and replied to from the switch from any interfaces or from any IP addresses.
`promiscuous`	Selecting this parameter results in the web authentication server on the switch intercepting and replying to any ARP or DNS messages from any IP address.

Default Intercept mode on the web authentication server is set to **none** by default.

Mode Global Configuration

Usage See the section "Web-authentication Enhancements" on page 39.10 in Chapter 39, Authentication Introduction and Configuration for further overview information about the Web-authentication enhancements, allowing Web-authentication to work as seamlessly as 802.1X authentication.

See the sub-sections "Interception of clients' ARPs" on page 39.10 and "Proxy DNS response" on page 39.11for an details of the associated usage of the available intercept modes.

See the section "Limitations on allowed feature combinations" on page 39.13 for information about restrictions regarding combinations of authentication enhancements working together.

Examples To enable the intercept mode on the web authentication server, resulting in the switch intercepting and replying to ARP and DNS messages from the same interface and IP address, use the following commands:

```
        awplus#  configure terminal
awplus(config)#  auth-web-server mode intercept
```

To disable the intercept mode on the web authentication server, use the following commands:

```
        awplus#  configure terminal
awplus(config)#  no auth-web-server mode intercept
```

To reset the intercept mode to the default setting of none on the web authentication server, use the following commands:

```
        awplus# configure terminal
awplus(config)# auth-web-server mode none
```

To enable the promiscuous mode on the web authentication server, resulting in the switch intercepting and replying to any ARP or DNS messages from any IP address, use the following commands:

```
        awplus# configure terminal
awplus(config)# auth-web-server mode promiscuous
```

To disable the promiscuous mode on the web authentication server, use the following commands:

```
        awplus# configure terminal
awplus(config)# no auth-web-server mode promiscuous
```

Validation Commands show running-config

Related Commands show auth-web-server

auth-web-server ping-poll enable

This command enables the ping polling to the supplicant (client device) that is authenticated by web authentication.

The **no** variant of this command disables the ping polling to the supplicant that is authenticated by web authentication.

Syntax

```
auth-web-server ping-poll enable
no auth-web-server ping-poll enable
```

Default The ping polling feature for web authentication is disabled by default.

Mode Global Configuration

Examples To enable the ping polling feature for web authentication, use the following commands:

```
        awplus# configure terminal
awplus(config)# auth-web-server ping-poll enable
```

To disable the ping polling feature for web authentication, use the following commands:

```
        awplus# configure terminal
awplus(config)# no auth-web-server ping-poll enable
```

Validation Commands

show auth-web
show auth-web-server
show running-config

auth-web-server ping-poll failcount

This command sets a fail count for the ping polling feature when used with web authentication. The **failcount** parameter specifies the number of unanswered pings. A supplicant (client device) is logged off when the number of unanswered pings are greater than the failcount set with this command.

Use the **no** variant of this command to resets the fail count for the ping polling feature to the default (5 pings).

Syntax

```
auth-web-server ping-poll failcount <1-100>

no auth-web-server ping-poll failcount
```

Parameter	Description
<1-100>	Count.

Default The default failcount for ping polling is 5 pings.

Mode Global Configuration

Examples To set the failcount of ping polling to 10 pings, use the following commands:

```
        awplus# configure terminal
awplus(config)# auth-web-server ping-poll failcount 10
```

To set the failcount of ping polling to default, use the following commands:

```
        awplus# configure terminal
awplus(config)# no auth-web-server ping-poll failcount
```

Validation Commands

show auth-web
show auth-web-server
show running-config

auth-web-server ping-poll interval

This command is used to change the ping poll interval. The interval specifies the time period between pings when the supplicant (client device) is reachable.

Use the **no** variant of this command to reset to the default period for ping polling (30 seconds).

Syntax

```
auth-web-server ping-poll interval <1-65535>
```

```
no auth-web-server ping-poll interval
```

Parameter	Description
<1-65535>	Seconds.

Default The interval for ping polling is 30 seconds by default.

Mode Global Configuration

Examples To set the interval of ping polling to 60 seconds, use the following commands:

```
        awplus# configure terminal
awplus(config)# auth-web-server ping-poll interval 60
```

To set the interval of ping polling to the default (30 seconds), use the following commands:

```
        awplus# configure terminal
awplus(config)# no auth-web-server ping-poll interval
```

Validation Commands

show auth-web
show auth-web-server
show running-config

auth-web-server ping-poll reauth-timer-refresh

This command modifies the **reauth-timer-refresh** parameter for the web-authentication feature. The **reauth-timer-refresh** parameter specifies whether a re-authentication timer is reset and when the response from a supplicant (a client device) is received.

Use the **no** variant of this command to reset the **reauth-timer-refresh** parameter to the default setting (disabled).

Syntax

```
auth-web-server ping-poll reauth-timer-refresh

no auth-web-server ping-poll reauth-timer-refresh
```

Default The `reauth-timer-refresh` parameter is disabled by default.

Mode Global Configuration

Examples To enable the `reauth-timer-refresh` timer, use the following commands:

```
        awplus# configure terminal
awplus(config)# auth-web-server ping-poll reauth-timer-refresh
```

To disable the `reauth-timer-refresh` timer, use the following commands:

```
        awplus# configure terminal
awplus(config)# no auth-web-server ping-poll reauth-timer-
                refresh
```

Validation Commands

show auth-web
show auth-web-server
show running-config

auth-web-server ping-poll timeout

This command modifies the ping poll **timeout** parameter for the web authentication feature. The **timeout** parameter specifies the time in seconds to wait for a response to a ping packet.

Use the **no** variant of this command to reset the timeout of ping polling to the default (1 second).

Syntax

```
auth-web-server ping-poll timeout <1-30>

no auth-web-server ping-poll timeout
```

Parameter	Description
<1-30>	Seconds.

Default The default timeout for ping polling is 1 second.

Mode Global Configuration

Examples To set the timeout of ping polling to 2 seconds, use the command:

```
        awplus#  configure terminal
awplus(config)#  auth-web-server ping-poll timeout 2
```

To set the timeout of ping polling to the default (1 second), use the command:

```
        awplus#  configure terminal
awplus(config)#  no auth-web-server ping-poll timeout
```

Validation Commands

show auth-web
show auth-web-server
show running-config

auth-web-server port

This command sets the HTTP port number for the web authentication server.

Use the **no** variant of this command to reset the HTTP port number to the default (80).

Syntax

```
auth-web-server port <port-number>

no auth-web-server port
```

Parameter	Description
<port-number>	Set the local web authentication server port within the TCP port number range 1 to 65535.

Default The web authentication server HTTP port number is set to 80 by default.

Mode Global Configuration

Examples To set the HTTP port number `8080` for the web authentication server, use the following commands:

```
        awplus# configure terminal
awplus(config)# auth-web-server port 8080
```

To reset to the default HTTP port number `80` for the web authentication server, use the following commands:

```
        awplus# configure terminal
awplus(config)# no auth-web-server port
```

Validation Commands

show auth-web
show auth-web-server
show running-config

auth-web-server redirect-url

This command sets a URL for supplicant (client device) authentication. When a supplicant is authorized it will be automatically redirected to the specified URL. Note that if the http redirect feature is used then this command is ignored.

Use the **no** variant of this command to delete the URL string set previously.

Syntax

```
auth-web-server redirect-url <url>

no auth-web-server redirect-url
```

Parameter	Description
<url>	URL (hostname or dotted IP notation).

Default The redirect URL for the web authentication server feature is not set by default (null).

Mode Global Configuration

Examples To enable and set redirect a URL string `www.alliedtelesis.com` for the web authentication server, use the following commands:

```
        awplus# configure terminal
awplus(config)# auth-web-server redirect-url
                http://www.alliedtelesis.com
```

To delete a redirect URL string, use the following commands:

```
        awplus# configure terminal
awplus(config)# no auth-web-server redirect-url
```

Validation Commands

show auth-web
show auth-web-server
show running-config

Related Commands auth-web-server http-redirect

auth-web-server session-keep

This command enables the session-keep feature to jump to the original URL after being authorized by web authentication.

Use the **no** variant of this command to disable the session keep feature.

Syntax

```
auth-web-server session-keep
no auth-web-server session-keep
```

Default The session-keep feature is disabled by default.

Mode Global Configuration

Examples To enable the session-keep feature, use the following commands:

```
        awplus# configure terminal
awplus(config)# auth-web-server session-keep
```

To disable the session-keep feature, use the following commands:

```
        awplus# configure terminal
awplus(config)# no auth-web-server session-keep
```

Validation Commands

show auth-web
show auth-web-server
show running-config

auth-web-server ssl

This command enables HTTPS functionality for the web authentication server feature.

Use the **no** variant of this command to disable HTTPS functionality for the web authentication server.

Syntax

```
auth-web-server ssl
```

```
no auth-web-server ssl
```

Default HTTPS functionality for the web authentication server feature is disabled by default.

Mode Global Configuration

Examples To enable HTTPS functionality for the web authentication server feature, use the following commands:

```
        awplus# configure terminal
awplus(config)# auth-web-server ssl
```

To disable HTTPS functionality for the web authentication server feature, use the following commands:

```
        awplus# configure terminal
awplus(config)# no auth-web-server ssl
```

Validation Commands

show auth-web
show auth-web-server
show running-config

auth-web-server sslport

This command sets the HTTPS port number for the web authentication server feature.

Use the **no** variant of this command to reset the HTTPS port number to the default port number (443) for the web authentication server feature.

Syntax

```
auth-web-server sslport <1-65535>

no auth-web-server sslport
```

Parameter	Description
<1-65535>	Set the local web authentication server port within the TCP port number range 1to 65535.

Default The HTTPS port number for the web authentication server feature is set to 443 by default.

Mode Global Configuration

Examples To set the HTTPS port number to 4433 for the web authentication server, use the command:

```
        awplus# configure terminal
awplus(config)# auth-web-server sslport 4433
```

To reset the HTTPS port number for the web authentication server to the default (443), use the command:

```
        awplus# configure terminal
awplus(config)# no auth-web-server sslport
```

Validation Commands

show auth-web
show auth-web-server
show running-config

copy web-auth-https-file

Use this command to download the SSL server certificate for web-based authentication. The file must be in PEM (Privacy Enhanced Mail) format, and contain the private key and the server certificate.

Syntax

```
copy <filename> web-auth-https-file
```

Parameter	Description
<filename>	The URL of the server certificate file.

Mode Privileged Exec

Example To download the server certificate file `veriSign_cert.pem` from the TFTP server directory `server`, use the command:

```
awplus# copy tftp://server/veriSign_cert.pem web-auth-https-
        file
```

Related Commands

auth-web-server ssl
erase web-auth-https-file
show auth-web-server

erase web-auth-https-file

Use this command to remove the SSL server certificate for web-based authentication.

Syntax

```
erase web-auth-https-file
```

Use this command to remove the SSL server certificate for web-based authentication.

Mode Privileged Exec

Example To remove the SSL server certificate file for web-based authentication use the command:

```
awplus# erase web-auth-https-file
```

Related Commands

auth-web-server ssl
copy web-auth-https-file
show auth-web-server

platform mac-vlan-hashing-algorithm

This command enables you to change the MAC VLAN hash-key-generating algorithm.

The no variant of this command returns the hash-key algorithm to 32l.

Syntax

```
platform mac-vlan-hashing-algorithm {crc16l|crc16u|crc32l|crc32u}

no platform mac-vlan-hashing-algorithm
```

Parameter	Description
`platform`	The global settings for the platform processor.
`mac-vlan-hashing-algorithm`	L2 MAC VLAN hash control.
`crc16l`	The algorithm that will apply to the lower bits of CRC-16
`crc16u`	The algorithm that will apply to the upper bits of CRC-16
The algorithm that will apply to the lower bits of CRC-32l	The algorithm that will apply to the lower bits of CRC-32
The algorithm that will apply to the upper bits of CRC-32u	The algorithm that will apply to the upper bits of CRC-32

Default 32l

Mode Config

Usage Occasionally, when using the Multiple Dynamic VLAN feature, a supplicant cannot be authenticated because a collision occurs within the VLAN MAC table. This can happen when more than four different MAC addresses produce the same hash-key.

A work-around when this situation occurs, can sometimes be applied by changing the hashing algorithm from its default of 32l. Several different algorithms may need to be tried to rectify the problem.

You must restart the switch for this command to take effect

Note that this command is intended for technical support staff, or advanced end users.

Example To change the hash-key generating algorithm applying to the lower bits of the CRC16l, use the command:

```
          awplus#  configure terminal
 awplus (config)#  platform mac-vlan-hashing-algorithm crc16l
```

Related Commands show platform

show auth-mac

This command shows authentication information for MAC-based authentication.

Syntax

```
show auth-mac [all]
```

Parameter	Description
`all`	Display all authentication information for each interface available on the switch.

Mode Privileged Exec

Example To display all MAC based authentication information, enter the command:

```
awplus# show auth-mac all
```

Output Figure 40-1: Example output from the **show auth-mac** command

```
802.1X Port-Based Authentication Disabled
MAC-based Port Authentication Enabled
WEB-based Port Authentication Disabled
```

Related Commands show dot1x
show auth-web

show auth-mac diagnostics

This command shows MAC authentication diagnostics, optionally for the specified interface, which may be a static channel (or static aggregator) or a dynamic (or LACP) channel group or a switch port.

If no interface is specified then authentication diagnostics are shown for all interfaces.

Syntax `show auth-mac diagnostics [interface <interface-list>]`

Parameter	Description
`interface`	Specify an interface to show
`<interface-list>`	The interfaces or ports to configure. An interface-list can be: ■ an interface (e.g. `vlan2`), a switch port (e.g. `port1.0.12`), a static channel group (e.g. `sa3`) or a dynamic (LACP) channel group (e.g. `po4`) ■ a continuous range of interfaces, ports, static channel groups or dynamic (LACP) channel groups separated by a hyphen; e.g. `vlan2-8`, or `port1.0.1-1.0.24`, or `sa2-4`, or `po1-3` ■ a comma-separated list of the above; e.g. `port1.0.1,port1.0.8-1.0.24`. Do not mix interface types in a list The specified interfaces must exist.

Mode Privileged Exec

Example To display authentication diagnostics for `port1.0.12`, enter the command:

```
awplus# show auth-mac diagnostics interface port1.0.12
```

Output Figure 40-2: Example output from the **show auth-mac diagnostics** command

```
Authentication Diagnostics for interface port1.0.12
    Supplicant address: 00d0.59ab.7037
        authEnterConnecting: 2
        authEaplogoffWhileConnecting: 1
        authEnterAuthenticating: 2
        authSuccessWhileAuthenticating: 1
        authTimeoutWhileAuthenticating: 1
        authFailWhileAuthenticating: 0
        authEapstartWhileAuthenticating: 0
        authEaplogoggWhileAuthenticating: 0
        authReauthsWhileAuthenticated: 0
        authEapstartWhileAuthenticated: 0
        authEaplogoffWhileAuthenticated: 0
        BackendResponses: 2
        BackendAccessChallenges: 1
        BackendOtherrequestToSupplicant: 3
        BackendAuthSuccess: 1
```

show auth-mac interface

This command shows the status for MAC based authentication on the specified interface, which may be a static channel (or static aggregator) or a dynamic (or LACP) channel group or a switch port.

Use the optional **diagnostics** parameter to show authentication diagnostics for the specified interface. Use the optional **sessionstatistics** parameter to show authentication session statistics for the specified interface. Use the optional **statistics** parameter to show authentication diagnostics for the specified interface. Use the optional **supplicant** (client device) parameter to show the supplicant state for the specified interface.

Syntax

```
show auth-mac interface <interface-list>
   [diagnostics|sessionstatistics|statistics|supplicant [brief]]
```

Parameter	Description
<interface-list>	The interfaces or ports to configure. An interface-list can be: ■ an interface (e.g. `vlan2`), a switch port (e.g. `port1.0.12`), a static channel group (e.g. `sa3`) or a dynamic (LACP) channel group (e.g. `po4`) ■ a continuous range of interfaces, ports, static channel groups or dynamic (LACP) channel groups separated by a hyphen; e.g. `vlan2-8`, or `port1.0.1-1.0.24`, or `sa2-4`, or `po1-3` ■ a comma-separated list of the above; e.g. `port1.0.1,port1.0.8-1.0.24`. Do not mix interface types in a list The specified interfaces must exist.
`diagnostics`	Diagnostics.
`sessionstatistics`	Session statistics.
`statistics`	Statistics.
`supplicant`	Supplicant (client device).
`brief`	Brief summary of supplicant state.

Mode Privileged Exec

Examples To display MAC based authentication status for `port1.0.12`, enter the command:

```
awplus# show auth-mac interface port1.0.2
```

```
% Port-Control not configured on port1.0.2
```

To display MAC authentication diagnostics for port1.0.12, enter the command:

```
awplus# show auth-mac interface port1.0.12 diagnostics
```

```
Authentication Diagnostics for interface port1.0.2
    Supplicant address: 00d0.59ab.7037
        authEnterConnecting: 2
        authEaplogoffWhileConnecting: 1
        authEnterAuthenticating: 2
        authSuccessWhileAuthenticating: 1
        authTimeoutWhileAuthenticating: 1
        authFailWhileAuthenticating: 0
        authEapstartWhileAuthenticating: 0
        authEaplogoggWhileAuthenticating: 0
        authReauthsWhileAuthenticated: 0
        authEapstartWhileAuthenticated: 0
        authEaplogoffWhileAuthenticated: 0
        BackendResponses: 2
        BackendAccessChallenges: 1
        BackendOtherrequestToSupplicant: 3
        BackendAuthSuccess: 1
```

To display authentication session statistics for port1.0.12, enter the command:

```
awplus# show auth-mac interface port1.0.12 sessionstatistics
```

```
Authentication session statistics for interface port1.0.12
    session user name: manager
        session authentication method: Remote server
        session time: 19440 secs
        session terminat cause: Not terminated yet
```

To display MAC authentication statistics for port1.0.12, enter the command:

```
awplus# show auth-mac interface port1.0.12 statistics
```

To display the MAC authenticated supplicant on interface port1.0.12, enter the command:

```
awplus# show auth-mac interface port1.0.12 supplicant
```

Related Commands

show auth-web diagnostics
show dot1x sessionstatistics
show dot1x statistics interface
show dot1x supplicant interface

show auth-mac sessionstatistics

This command shows authentication session statistics for the specified interface, which may be a static channel (or static aggregator) or a dynamic (or LACP) channel group or a switch port.

Syntax `show auth-mac sessionstatistics [interface <interface-list>]`

Parameter	Description
`interface`	Specify an interface to show.
`<interface-list>`	The interfaces or ports to configure. An interface-list can be: ■ an interface (e.g. `vlan2`), a switch port (e.g. `port1.0.12`), a static channel group (e.g. `sa3`) or a dynamic (LACP) channel group (e.g. `po4`) ■ a continuous range of interfaces, ports, static channel groups or dynamic (LACP) channel groups separated by a hyphen; e.g. `vlan2-8`, or `port1.0.1-1.0.24`, or `sa2-4`, or `po1-3` ■ a comma-separated list of the above; e.g. `port1.0.1,port1.0.8-1.0.24`. Do not mix interface types in a list The specified interfaces must exist.

Mode Privileged Exec

Example To display output displaying MAC authentication session statistics for `port1.0.12`, enter the command:

```
awplus# show auth-mac sessionstatistics interface port1.0.12
```

Output Figure 40-3: Example output from the **show auth-mac sessionstatistics** command

```
Authentication session statistics for interface port1.0.12
    session user name: manager
        session authentication method: Remote server
        session time: 19440 secs
        session terminat cause: Not terminated yet
```

show auth-mac statistics interface

This command shows the authentication statistics for the specified interface, which may be a static channel (or static aggregator) or a dynamic (or LACP) channel group or a switch port.

Syntax

```
show auth-mac statistics [interface <interface-list>]
```

Parameter	Description
`interface`	Specify ports to show.
`<interface-list>`	The interfaces or ports to configure. An interface-list can be: ■ an interface (e.g. `vlan2`), a switch port (e.g. `port1.0.12`), a static channel group (e.g. `sa3`) or a dynamic (LACP) channel group (e.g. `po4`) ■ a continuous range of interfaces, ports, static channel groups or dynamic (LACP) channel groups separated by a hyphen; e.g. `vlan2-8`, or `port1.0.1-1.0.24`, or `sa2-4`, or `po1-3` ■ a comma-separated list of the above; e.g. `port1.0.1,port1.0.8-1.0.24`. Do not mix interface types in a list The specified interfaces must exist.

Mode Privileged Exec

Example To display MAC authentication statistics for `port1.0.12`, enter the command:

```
awplus# show auth-mac statistics interface port1.0.12
```

Related Commands show dot1x interface

show auth-mac supplicant

This command shows the supplicant (client device) state when MAC authentication is configured for the switch. This command shows a summary when the optional **brief** parameter is used.

Syntax
```
show auth-mac supplicant [<macadd>] [brief]
```

Parameter	Description
`<macadd>`	Mac (hardware) address of the Supplicant Entry format is HHHH.HHHH.HHHH (hexadecimal).
`brief`	Brief summary of the Supplicant state.

Mode Privileged Exec

Example To display the MAC authenticated supplicant for MAC address `00d0.59ab.7037`, enter the command:

```
awplus# show auth-mac supplicant 00d0.59ab.7037
```

show auth-mac supplicant interface

This command shows the supplicant (client device) state for the MAC authenticated interface, which may be a static channel (or static aggregator) or a dynamic (or LACP) channel group or a switch port. This command shows a summary when the optional **brief** parameter is used.

Syntax `show auth-mac supplicant [interface <interface-list>] [brief]`

Parameter	Description
`interface`	Specify ports to show.
`<interface-list>`	The interfaces or ports to configure. An interface-list can be: ■ an interface (e.g. `vlan2`), a switch port (e.g. `port1.0.12`), a static channel group (e.g. `sa3`) or a dynamic (LACP) channel group (e.g. `po4`) ■ a continuous range of interfaces, ports, static channel groups or dynamic (LACP) channel groups separated by a hyphen; e.g. `vlan2-8`, or `port1.0.1-1.0.24`, or `sa2-4`, or `po1-3` ■ a comma-separated list of the above; e.g. `port1.0.1,port1.0.8-1.0.24`. Do not mix interface types in a list The specified interfaces must exist.
`brief`	Brief summary of the supplicant state.

Mode Privileged Exec

Examples To display the MAC authenticated supplicant on the interface `port1.0.12`, enter the command:

```
awplus# show auth-mac supplicant interface port1.0.12
```

To display brief summary output for the MAC authenticated supplicant, enter the command:

```
awplus# show auth-mac supplicant brief
```

show auth-web

This command shows authentication information for Web-based authentication.

Syntax

```
show auth-web [all]
```

Parameter	Description
`all`	Display all authentication information for each authenticated interface, which may be a static channel (or static aggregator) or a dynamic (or LACP) channel group or a switch port, available on the switch.

Mode Privileged Exec

Example To display all Web authentication information, enter the command:

```
awplus# show auth-web all
```

Output Figure 40-4: Example output from the **show auth-web** command

```
802.1X Port-Based Authentication Disabled
MAC-based Port Authentication Disabled
WEB-based Port Authentication Enabled
```

Related Commands show dot1x
show auth-mac

show auth-web diagnostics

This command shows Web authentication diagnostics, optionally for the specified interface, which may be a static channel (or static aggregator) or a dynamic (or LACP) channel group or a switch port.

If no interface is specified then authentication diagnostics are shown for all interfaces.

Syntax `show auth-web diagnostics [interface <interface-list>]`

Parameter	Description
`interface`	Specify ports to show.
`<interface-list>`	The interfaces or ports to configure. An interface-list can be: ■ an interface (e.g. `vlan2`), a switch port (e.g. `port1.0.12`), a static channel group (e.g. `sa3`) or a dynamic (LACP) channel group (e.g. `po4`) ■ a continuous range of interfaces, ports, static channel groups or dynamic (LACP) channel groups separated by a hyphen; e.g. `vlan2-8`, or `port1.0.1-1.0.24`, or `sa2-4`, or `po1-3` ■ a comma-separated list of the above; e.g. `port1.0.1,port1.0.8-1.0.24`. Do not mix interface types in a list The specified interfaces must exist.

Mode Privileged Exec

Example To display authentication diagnostics for `port1.0.12`, enter the command:

```
awplus# show auth-web diagnostics interface port1.0.12
```

Output Figure 40-5: Example output from the **show auth-web diagnostics** command

```
Authentication Diagnostics for interface port1.0.12
    Supplicant address: 00d0.59ab.7037
        authEnterConnecting: 2
        authEaplogoffWhileConnecting: 1
        authEnterAuthenticating: 2
        authSuccessWhileAuthenticating: 1
        authTimeoutWhileAuthenticating: 1
        authFailWhileAuthenticating: 0
        authEapstartWhileAuthenticating: 0
        authEaplogoggWhileAuthenticating: 0
        authReauthsWhileAuthenticated: 0
        authEapstartWhileAuthenticated: 0
        authEaplogoffWhileAuthenticated: 0
        BackendResponses: 2
        BackendAccessChallenges: 1
        BackendOtherrequestToSupplicant: 3
        BackendAuthSuccess: 1
```

Related Commands show dot1x interface

show auth-web interface

This command shows the status for Web based authentication on the specified interface, which may be a static channel (or static aggregator) or a dynamic (or LACP) channel group or a switch port.

Use the optional **diagnostics** parameter to show authentication diagnostics for the specified interface. Use the optional **sessionstatistics** parameter to show authentication session statistics for the specified interface. Use the optional **statistics** parameter to show authentication diagnostics for the specified interface. Use the optional **supplicant** (client device) parameter to show the supplicant state for the specified interface.

Syntax

```
show auth-web interface <interface-list>
   [diagnostics|sessionstatistics|statistics|supplicant [brief]]
```

Parameter	Description
<interface-list>	The interfaces or ports to configure. An interface-list can be: ■ an interface (e.g. `vlan2`), a switch port (e.g. `port1.0.12`), a static channel group (e.g. `sa3`) or a dynamic (LACP) channel group (e.g. `po4`) ■ a continuous range of interfaces, ports, static channel groups or dynamic (LACP) channel groups separated by a hyphen; e.g. `vlan2-8`, or `port1.0.1-1.0.24`, or `sa2-4`, or `po1-3` ■ a comma-separated list of the above; e.g. `port1.0.1,port1.0.8-1.0.24`. Do not mix interface types in a list The specified interfaces must exist.
`diagnostics`	Diagnostics.
`sessionstatistics`	Session statistics.
`statistics`	Statistics.
`supplicant`	Supplicant (client device).
`brief`	Brief summary of supplicant state.

Mode Privileged Exec

Example To display the Web based authentication status for `port1.0.12`, enter the command:

```
awplus# show auth-web interface port1.0.2
```

```
% Port-Control not configured on port1.0.2
```

To display Web authentication diagnostics for `port1.0.12`, enter the command:

```
awplus# show auth-web interface port1.0.12 diagnostics
```

```
Authentication Diagnostics for interface port1.0.2
    Supplicant address: 00d0.59ab.7037
        authEnterConnecting: 2
        authEaplogoffWhileConnecting: 1
        authEnterAuthenticating: 2
        authSuccessWhileAuthenticating: 1
        authTimeoutWhileAuthenticating: 1
        authFailWhileAuthenticating: 0
        authEapstartWhileAuthenticating: 0
        authEaplogoggWhileAuthenticating: 0
        authReauthsWhileAuthenticated: 0
        authEapstartWhileAuthenticated: 0
        authEaplogoffWhileAuthenticated: 0
        BackendResponses: 2
        BackendAccessChallenges: 1
        BackendOtherrequestToSupplicant: 3
        BackendAuthSuccess: 1
```

To display Web authentication session statistics for `port1.0.12`, enter the command:

```
awplus# show auth-web interface port1.0.12 sessionstatistics
```

```
Authentication session statistics for interface port1.0.12
    session user name: manager
        session authentication method: Remote server
        session time: 19440 secs
        session terminat cause: Not terminated yet
```

To display Web authentication statistics for `port1.0.12`, enter the command:

```
awplus# show auth-web statistics interface port1.0.12
```

To display the Web authenticated supplicant on interface `port1.0.12`, enter the command:

```
awplus# show auth-web interface port1.0.12 supplicant
```

Related Commands

show auth-web diagnostics
show dot1x sessionstatistics
show dot1x statistics interface
show dot1x supplicant interface

show auth-web sessionstatistics

This command shows authentication session statistics for the specified interface, which may be a static channel (or static aggregator) or a dynamic (or LACP) channel group or a switch port.

Syntax `show auth-web sessionstatistics [interface <interface-list>]`

Parameter	Description
`interface`	Specify ports to show.
`<interface-list>`	The interfaces or ports to configure. An interface-list can be: ■ an interface (e.g. `vlan2`), a switch port (e.g. `port1.0.12`), a static channel group (e.g. `sa3`) or a dynamic (LACP) channel group (e.g. `po4`) ■ a continuous range of interfaces, ports, static channel groups or dynamic (LACP) channel groups separated by a hyphen; e.g. `vlan2-8`, or `port1.0.1-1.0.24`, or `sa2-4`, or `po1-3` ■ a comma-separated list of the above; e.g. `port1.0.1,port1.0.8-1.0.24`. Do not mix interface types in a list The specified interfaces must exist.

Mode Privileged Exec

Example To display authentication statistics for `port1.0.12`, enter the command:

```
awplus# show auth-web sessionstatistics interface port1.0.12
```

Output Figure 40-6: Example output from the **show auth-web sessionstatistics** command

```
Authentication session statistics for interface port1.0.12
    session user name: manager
        session authentication method: Remote server
        session time: 19440 secs
        session terminat cause: Not terminated yet
```

show auth-web statistics interface

This command shows the authentication statistics for the specified interface, which may be a static channel (or static aggregator) or a dynamic (or LACP) channel group or a switch port.

Syntax

```
show auth-web statistics interface <interface-list>
```

Parameter	Description
<interface-list>	The interfaces or ports to configure. An interface-list can be: ■ an interface (e.g. `vlan2`), a switch port (e.g. `port1.0.12`), a static channel group (e.g. `sa3`) or a dynamic (LACP) channel group (e.g. `po4`) ■ a continuous range of interfaces, ports, static channel groups or dynamic (LACP) channel groups separated by a hyphen; e.g. `vlan2-8`, or `port1.0.1-1.0.24`, or `sa2-4`, or `po1-3` ■ a comma-separated list of the above; e.g. `port1.0.1,port1.0.8-1.0.24`. Do not mix interface types in a list The specified interfaces must exist.

Mode Privileged Exec

Example To display Web authentication statistics for `port1.0.12`, enter the command:

```
awplus# show dot1x statistics interface port1.0.12
```

Related Commands show dot1x interface

show auth-web supplicant

This command shows the supplicant (client device) state when Web authentication is configured for the switch. This command shows a summary when the optional **brief** parameter is used.

Syntax

```
show auth-web supplicant [<macadd>] [brief]
```

Parameter	Description
`<macadd>`	Mac (hardware) address of the supplicant Entry format is HHHH.HHHH.HHHH (hexadecimal).
`brief`	Brief summary of the supplicant state.

Mode Privileged Exec

Examples To display Web authenticated supplicant information on the switch, enter the command:

```
awplus# show auth-web supplicant
```

To display brief summary output for the Web authenticated supplicant on the switch, enter the command:

```
awplus# show auth-web supplicant brief
```

show auth-web supplicant interface

This command shows the supplicant (client device) state for the Web authenticated interface, which may be a static channel (or static aggregator) or a dynamic (or LACP) channel group or a switch port. This command shows a summary when the optional **brief** parameter is used.

Syntax

```
show auth-web supplicant interface <interface-list> [brief]
```

Parameter	Description
`<interface-list>`	The interfaces or ports to configure. An interface-list can be: ■ an interface (e.g. `vlan2`), a switch port (e.g. `port1.0.12`), a static channel group (e.g. `sa3`) or a dynamic (LACP) channel group (e.g. `po4`) ■ a continuous range of interfaces, ports, static channel groups or dynamic (LACP) channel groups separated by a hyphen; e.g. `vlan2-8`, or `port1.0.1-1.0.24`, or `sa2-4`, or `po1-3` ■ a comma-separated list of the above; e.g. `port1.0.1,port1.0.8-1.0.24`. Do not mix interface types in a list The specified interfaces must exist.
`brief`	Brief summary of the supplicant state.

Mode Privileged Exec

Examples To display the Web authenticated supplicant on the interface `port1.0.12`, enter the command:

```
awplus# show auth-web supplicant interface port1.0.12
```

To display brief summary output for the Web authenticated supplicant, enter the command:

```
awplus# show auth-web supplicant brief
```

show auth-web-server

This command shows the web authentication server configuration and status on the switch.

Syntax

```
show auth-web-server
```

Mode

Privileged Exec

Example

To display web authentication server configuration and status, enter the command:

```
awplus# show auth-web-server
```

Output

Figure 40-7: Example output from the **show auth-web-server** command

```
Web authentication server
    Server status: enabled
    Server address: -
    HTTP Port No: 80
    Security: enabled
    Certification: default
    SSL Port No: 443
    Redirect URL:
    HTTP Redirect: disabled
    Session keep: disabled
    PingPolling: disable
    PingInterval: 30
    Timeout: 1
    FailCount: 5
    ReauthFresh: disabled
```

Related Commands

auth-web-server http-redirect
auth-web-server ipaddress
auth-web-server port
auth-web-server redirect-url
auth-web-server session-keep
auth-web-server ssl
auth-web-server sslport

Chapter 41: AAA Introduction and Configuration

AAA Introduction

AAA is the collective title for the three related functions of Authentication, Authorization and Accounting. These function can be applied in a variety of methods with a variety of servers. The purpose of the AAA commands is to map instances of the AAA functions to sets of servers.

The Authentication function can be performed in multiple contexts, such as authentication of users logging in at a console, or 802.1x authentication of devices connecting to Ethernet ports.

For each of these contexts, you may want to use different sets of servers for examining the proffered authentication credentials and deciding if they are valid. AAA Authentication commands enable you to specify which servers will be used for different types of authentication.

Available functions and server types

Authentication, Authorization and Accounting functions are available.

Authentication is performed in the following contexts:

- Login authentication of user shell sessions on the console port, and via telnet/SSH
- Enable password authentication for user shell sessions on the console port, and via telnet/SSH (TACACS+ only)
- 802.1x authentication of devices connecting to switch ports
- MAC authentication of devices connecting to switch ports
- Web-based authentication of devices connecting to switch ports

Authorization is performed in the following context:

- TACACS+ login authentication. Note that with the AlliedWare Plus TACACS+ implementation:
  - authorization cannot be performed independently of the login authentication process
  - authorization will not be attempted if enable password authentication is configured
  - there are no authorization commands available

Accounting is performed in the following contexts:

- Accounting of console, telnet, and SSH login sessions
- Accounting of commands executed within user shell sessions (TACACS+ only)
- Accounting of 802.1x-authenticated connections
- Accounting of MAC-authenticated connections
- Accounting of Web-authenticated connections

The three types of servers that can be used are:

- Local user database
- RADIUS servers
- TACACS+ servers

Server Groups and Method Lists

There are two constructs that underlie the structure of the AAA commands:

- Server groups are lists of RADIUS servers
- Method Lists are lists of server types

Server Groups

A server group is defined by the command aaa group server. This command puts you into Server Group configuration mode. Once in that mode you can add servers to the group by using the command server auth-port.

Any number of servers can be added to a group. Typically, you will add servers which have already been configured by the command radius-server host. If you add a server that has not yet been configured by the command radius-server host, you will receive a warning that the server has not yet been configured, but the command will be accepted.

There is one server group that is always present on the switch by default that cannot be removed. It is the group simply named **radius** that comprises all servers that have been configured using the command radius-server host. As soon as a server is configured by the command radius-server host, it is automatically a member of the server group **radius** and cannot be removed from it.

Method Lists

A method list defines the set of server types that you want to be used for authenticating a user/ device, and the order in which you want the server types to be used.

- You may want the usernames proffered for logging in at the console to be checked for in the local user database. You can create a server list that specifies **local**.
- You may want to check the TACACS+ servers first, and resort to the local user database if none of the TACACS+ servers respond. You can create a server list that specifies **group tacacs+** first, followed by **local**.
- You may want to check the RADIUS servers first, and resort to the local user database if none of the RADIUS servers respond. You can create a server list that specifies **group radius** first, followed by **local**.

A method list defines the servers where authentication requests are sent. The first server listed is used to authenticate users; if that server fails then the next authentication server type in the method list is selected. This process continues until there is a successful authentication or until all server types fail.

When a user attempts to log in, the switch sends an authentication request to the first authentication server in the method list. If the first server in the list is reachable and it contains a username and password matching the authentication request, the user is authenticated and the login succeeds. If the authentication server denies the authentication request because of an incorrect username or password, the user login fails. If the first server in the method list is unreachable, the switch sends the request to the next server in the list, and so on.

For example, if the method list specifies `group tacacs+ local`, and a user attempts to log in with a password that does not match a user entry in the first TACACS+ server, if this TACACS+ server denies the authentication request, then the switch does not try any other TACACS+ servers not the local user database; the user login fails.

Default Method Lists

For every authentication or accounting type, it is always possible to define a method list called **default**. For most of the authentication and accounting types, the only method list that can be defined is default.

As soon as the default method list is defined for a given authentication or accounting type, it is automatically applied as the method list to be used for any instance of that type of authentication or accounting, except for instances to which another named method list has been specifically applied.

Configuring AAA Login Authentication

To configure AAA authentication, create default or a named method list for different authentication types. In the case of login authentication, the named method lists are then applied to consoles or VTY lines.

AAA Configuration Tasks

To define how a given accounting or authentication type will be applied to a given port or line:

- either create a server group using the aaa group server command (RADIUS only),
- or create a method list for the authentication or accounting type as required,
- then apply that method list to the port or line as required.

Step 1: Define a group of RADIUS Servers:

Create a server group using the aaa group server command.

To create a RADIUS server group named `GROUP1` with hosts `192.168.1.1`, `192.168.2.1` and `192.168.3.1`, use the commands:

```
   awplus(config)# aaa group server radius GROUP1
awplus(config-sg)# server 192.168.1.1 auth-port 1812 acct-
                   port 1813
awplus(config-sg)# server 192.168.2.1 auth-port 1812 acct-
                   port 1813
awplus(config-sg)# server 192.168.3.1 auth-port 1812 acct-
                   port 1813
```

Step 2: Specify the login authentication or accounting Method List:

Create a method list for the authentication (aaa authentication login) or accounting (aaa accounting login) type as required.

To configure a user login authentication method list called `USERS` to use first all available RADIUS servers for user login authentication and then the local user database, use the following commands:

```
        awplus# configure terminal
awplus(config)# aaa authentication login default group radius
                local
```

To configure RADIUS accounting for login shell sessions, use the following commands:

```
        awplus# configure terminal
awplus(config)# aaa accounting login default start-stop group
                radius
```

To configure a user login authentication method list called USERS to use first the TACACS+ servers for user login authentication and then the local user database, use the following commands:

```
        awplus#  configure terminal
awplus(config)#  aaa authentication login USERS group tacacs+
                 local
```

Step 3: Apply Method List to Interface Port or Line:

Apply that method list to the port or line as required.

```
             awplus#  configure terminal
     awplus(config)#  line console 0
awplus(config-line)#  login authentication USERS
```

For most Authentication and Accounting types, the only possible server list is **default**, and the only server that can be put into it is **radius**. You will typically use all RADIUS servers, so **group radius** can be used, rather than having to create a specific user group. Often the configuration of a given Authentication or Accounting type will consist of a single command, the command that defines the default server list, which contains just one server.

AAA 802.1x Authentication Configuration:

AAA 802.1x authentication will typically be configured with the following commands.

To enable 802.1x Authentication globally for all RADIUS servers, use the following commands:

```
        awplus#  configure terminal
awplus(config)#  aaa authentication dot1x default group radius
```

Sample Authentication Configurations

Sample 802.1X Authentication Configuration

See the below sample configuration script for a sample 802.1X authentication configuration. Copy and paste then edit the sample 802.1X authentication configuration in your config file. See the edit command in the Chapter 7, File Management Commands for further information.

Output

Figure 41-1: Sample 802.1X Authentication Configuration

```
!
 radius-server host 127.0.0.1 key awplus-local-radius-server
!
 aaa authentication dot1x default group radius
!
 radius-server local
 server enable
 nas 127.0.0.1 key awplus-local-radius-server
 user guest password guest!
!
 no spanning-tree rstp enable
!
 interface port1.0.1
 switchport
 switchport mode access
 dot1x port-control auto
!
 interface vlan1
 ip address 192.168.1.120/24
!
```

The 802.1X authentication feature needs the aaa authentication dot1x command and the dot1x port-control command configured on an interface. See Chapter 42, AAA Commands and Chapter 38, 802.1X Commands for command information to edit this configuration.

Local RADIUS Server has been configured to use 802.1X authentication in this sample configuration. See the radius-server local and server enable commands in Chapter 48, Local RADIUS Server Commands for command information to edit this sample configuration.

This sample configuration enables 802.1X authentication on interface `vlan1` with IP address `192.168.1.120`. Change the VLAN ID and IP address as required for your configuration.

Sample MAC Authentication Configuration

See the below sample configuration script for a sample MAC authentication configuration. Copy, paste, and edit the sample MAC authentication configuration in the config file. See the edit command in the Chapter 7, File Management Commands for further information.

Output

Figure 41-2: Sample MAC Authentication Configuration

```
!
 radius-server host 127.0.0.1 key awplus-local-radius-server
!
 aaa authentication auth-mac default group radius
!
 radius-server local
 server enable
 nas 127.0.0.1 key awplus-local-radius-server
 user 00-d0-59-ab-70-37 password 00-d0-59-ab-70-37
!
 no spanning-tree rstp enable
!
 interface port1.0.1
 switchport
 switchport mode access
 auth-mac enable
!
 interface vlan1
 ip address 192.168.1.120/24
!
```

The MAC authentication feature needs the aaa authentication auth-mac and the auth-mac enable commands configured on an interface. See Chapter 42, AAA Commands and Chapter 40, Authentication Commands for command information to edit this configuration.

Local RADIUS Server has been configured to use MAC authentication in this sample configuration. See the radius-server local and server enable commands in Chapter 48, Local RADIUS Server Commands for command information to edit this sample configuration.

See the user (RADIUS server) command in Chapter 48, Local RADIUS Server Commands for command information to edit the MAC address of the supplicant for use with local RADIUS server as the RADIUS user name and the user password, as shown in the above configuration.

This configuration enables MAC authentication on `vlan1` with IP address `192.168.1.120`. Change the interface VLAN ID, MAC, and IP addresses as needed in your configuration.

Sample Web-Authentication Configuration

See the below sample configuration script for a sample Web-authentication configuration. Copy, paste, and edit the sample Web-authentication configuration for your config file. See the edit command in the Chapter 7, File Management Commands for further information.

Output

Figure 41-3: Sample Web-Authentication Configuration

```
!
 radius-server host 127.0.0.1 key awplus-local-radius-server
!
 aaa authentication auth-web default group radius
!
 radius-server local
 server enable
 nas 127.0.0.1 key awplus-local-radius-server
 user guest encrypted password
 l+lWcLjLm29bCAXwWRPHXK0PFlsA7gNpR+P7wO4kwQQ=
!
 no spanning-tree rstp enable
!
 interface port1.0.1
 switchport
 switchport mode access
 auth-web enable
!
 interface vlan1
 ip address 192.168.1.120/24
!
```

The Web-authentication feature needs the aaa authentication auth-web and the auth-web enable commands configured on an interface. See Chapter 42, AAA Commands and Chapter 40, Authentication Commands for command information to edit this configuration.

Local RADIUS Server has been configured to use Web-authentication in this sample configuration. See the radius-server local and server enable commands in Chapter 48, Local RADIUS Server Commands for command information to edit this sample configuration.

The above sample Web-authentication configuration requires the user name ‘`guest`’ with password ‘`guest!`’ on IP address `192.168.1.120` from interface `port1.0.1`.

Sample Tri-Authentication Configuration

See the below sample configuration script for a sample tri-authentication configuration that configures 802.1X authentication, MAC authentication, and Web-authentication on the same interface. Copy, paste, and edit the sample tri-authentication configuration for your config file. See the edit command in the Chapter 7, File Management Commands for further information.

Output

Figure 41-4: Sample Tri-Authentication Configuration

```
!
 radius-server host 127.0.0.1 key awplus-local-radius-server
!
 aaa authentication dot1x default group radius
 aaa authentication auth-mac default group radius
 aaa authentication auth-web default group radius
!
 radius-server local
 server enable
 nas 127.0.0.1 key awplus-local-radius-server
 user guest password guest!
 user 00-d0-59-ab-70-37 password 00-d0-59-ab-70-37
!
 no spanning-tree rstp enable
!
 interface port1.0.1
 switchport
 switchport mode access
 dot1x port-control auto
 auth-mac enable
 auth-web enable
!
 interface vlan1
 ip address 192.168.1.120/24
!
```

The 802.1X authentication feature needs the aaa authentication dot1x command and the dot1x port-control command configured on an interface. See Chapter 42, AAA Commands and Chapter 38, 802.1X Commands for command information to edit this configuration.

The MAC authentication feature needs the aaa authentication auth-mac and the auth-mac enable commands configured on an interface. See Chapter 42, AAA Commands and Chapter 40, Authentication Commands for command information to edit this configuration.

The Web-authentication feature needs the aaa authentication auth-web and the auth-web enable commands configured on an interface. See Chapter 42, AAA Commands and Chapter 40, Authentication Commands for command information to edit this configuration.

Local RADIUS Server has been configured to use tri-authentication in this sample configuration. See the radius-server local and server enable commands in Chapter 48, Local RADIUS Server Commands for command information to edit this sample configuration.

This sample tri-authentication configuration requires a user name '`guest`' with password '`guest!`' on IP address `192.168.1.120` from `port1.0.1`. Note this sample also configures 802.1X and MAC authentication on `vlan1` with IP address `192.168.1.120`. Change the interface VLAN ID, MAC and IP address as needed for your configuration.

Note that when tri-authentication is applied to the same interface then the order of execution is MAC authentication first, then 802.1X or Web-authentication, if MAC authentication fails.

Chapter 42: AAA Commands

Command List

This chapter provides an alphabetical reference for AAA commands for Authentication, Authorization and Accounting. For more information, see Chapter 41, AAA Introduction and Configuration.

aaa accounting auth-mac default

This command configures a default accounting method list for MAC-based Authentication. The default accounting method list specifies what type of accounting messages are sent and specifies which RADIUS Servers the accounting messages are sent to. The default accounting method list is automatically applied to interfaces with MAC-based Authentication enabled.

Use the **no** variant of this command to disable AAA accounting for MAC-based Authentication globally.

Syntax

```
aaa accounting auth-mac default {start-stop|stop-only|none}
    group {<group-name>|radius}
```

```
no aaa accounting auth-mac default
```

Parameter	Description
`start-stop`	Start and stop records to be sent.
`stop-only`	Stop records to be sent.
`none`	No accounting record to be sent.
`<group-name>`	Server group name.
`radius`	Use all RADIUS servers

Default RADIUS accounting for MAC-based Authentication is disabled by default

Mode Global Configuration

Usage There are two ways to define servers where RADIUS accounting messages are sent:

- **group radius**: use all RADIUS servers configured by radius-server host command
- **group <*group-name*>**: use the specified RADIUS server group configured with the aaa group server command

The accounting event to send to the RADIUS server is configured with the following options:

- **start-stop**: sends a **start** accounting message at the beginning of a session and a **stop** accounting message at the end of the session.
- **stop-only**: sends a **stop** accounting message at the end of a session.
- **none**: disables accounting.

Examples To enable RADIUS accounting for MAC-based Authentication, and use all available RADIUS Servers, use the commands:

```
        awplus# configure terminal
awplus(config)# aaa accounting auth-mac default start-stop
                group radius
```

To disable RADIUS accounting for MAC-based Authentication, use the commands:

```
        awplus# configure terminal
awplus(config)# no aaa accounting auth-mac default
```

Related Commands aaa authentication auth-mac

aaa accounting auth-web default

This command configures a default accounting method list for Web-based Port Authentication. The default accounting method list specifies what type of accounting messages are sent and specifies which RADIUS Servers the accounting messages are sent to. The default accounting method list is automatically applied to interfaces with Web-based Authentication enabled.

Use the **no** variant of this command to disable AAA accounting for Web-based Port Authentication globally.

Syntax

```
aaa accounting auth-web default {start-stop|stop-only|none}
    group {<group-name>|radius}

no aaa accounting auth-web default
```

Parameter	Description
`start-stop`	Start and stop records to be sent.
`stop-only`	Stop records to be sent.
`none`	No accounting record to be sent.
`<group-name>`	Server group name.
`radius`	Use all RADIUS servers.

Default RADIUS accounting for WEB-based Port Authentication is disabled by default.

Mode Global Configuration

Usage There are two ways to define servers where RADIUS accounting messages are sent:

- **group radius**: use all RADIUS servers configured by radius-server host command
- **group <*group-name*>**: use the specified RADIUS server group configured with the aaa group server command

Configure the accounting event to be sent to the RADIUS server with the following options:

- **start-stop**: sends a **start** accounting message at the beginning of a session and a **stop** accounting message at the end of the session.
- **stop-only**: sends a **stop** accounting message at the end of a session.
- **none**: disables accounting.

Examples To enable RADIUS accounting for Web-based Authentication, and use all available RADIUS Servers, use the commands:

```
        awplus# configure terminal
awplus(config)# aaa accounting auth-web default start-stop
                group radius
```

To disable RADIUS accounting for Web-based Authentication, use the commands:

```
        awplus# configure terminal
awplus(config)# no aaa accounting auth-web default
```

Related Commands aaa authentication auth-web

aaa accounting commands

Use this command to configure and enable TACACS+ command accounting. When command accounting is enabled, information about a command entered at a specified privilege level on a device is sent to a TACACS+ server. To account for all commands entered on a device you need to configure command accounting for each discrete privilege level. A command accounting record includes the command as entered for the specified privilege level, the date and time each command execution finished, and the username of the user who executed the command.

This command creates a default method list that is applied to every console and vty line. The **stop-only** parameter indicates that an accounting message is sent to the TACACS+ server when a command has stopped executing.

Note that up to four TACACS+ servers can be configured for accounting. The servers are checked for reachability in the order they are configured and only the first reachable server is used. If no server is found the accounting message is dropped.

Use the **no** variant of this command to disable command accounting.

Syntax

```
aaa accounting commands <1-15> default stop-only group tacacs+
```

```
no aaa accounting commands <1-15> default
```

Parameter	Description
<1-15>	The privilege level, in the range 1 to 15.

Default TACACS+ command accounting is disabled by default.

Mode Global Configuration

Usage When command accounting is enabled, the command as entered is included in the accounting packets sent to the TACACS+ accounting server.

You cannot enable command accounting if a trigger is configured. An error message is displayed if you attempt to enable command accounting and a trigger is configured.

The show tech-support command runs a number of commands and each command is accounted separately.

When the **copy <*filename*> running-config** command is executed all the commands of a configuration file copied into the running-config are accounted separately.

Examples To configure command accounting for privilege level 15 commands, use the following commands:

```
        awplus# configure terminal
awplus(config)# aaa accounting commands 15 default stop-only
                group tacacs+
```

To disable command accounting for privilege level 15 commands, use the following commands:

```
        awplus# configure terminal
awplus(config)# no aaa accounting commands 15 default
```

Related Commands aaa authentication login
aaa accounting login
accounting login
tacacs-server host

aaa accounting dot1x

This command configures the default accounting method list for IEEE 802.1x-based Authentication. The default accounting method list specifies what type of accounting messages are sent and specifies which RADIUS Servers the accounting messages are sent to. The default accounting method list is automatically applied to interfaces with IEEE 802.1x-based Authentication enabled.

Use the **no** variant of this command to disable AAA accounting for 802.1x-based Port Authentication globally.

Syntax

```
aaa accounting dot1x default {start-stop|stop-only|none}
    group {<group-name>|radius}

no aaa accounting dot1x default
```

Parameter	Description
start-stop	Start and stop records to be sent.
stop-only	Stop records to be sent.
none	No accounting record to be sent.
<group-name>	Server group name.
radius	Use all RADIUS servers.

Default RADIUS accounting for 802.1X-based Port Authentication is disabled by default. (There is no default server set by default).

Mode Global Configuration

Usage There are two ways to define servers where RADIUS accounting messages will be sent:

- **group radius**: use all RADIUS servers configured by radius-server host command
- **group <*group-name*>**: use the specified RADIUS server group configured with the aaa group server command

The accounting event to send to the RADIUS server is configured by the following options:

- **start-stop**: sends a **start** accounting message at the beginning of a session and a **stop** accounting message at the end of the session.
- **stop-only**: sends a **stop** accounting message at the end of a session.
- **none**: disables accounting.

Example To enable RADIUS accounting for 802.1x-based Authentication, and use all available RADIUS Servers, use the commands:

```
        awplus# configure terminal
awplus(config)# aaa accounting dot1x default start-stop group
                radius
```

To disable RADIUS accounting for 802.1x-based Authentication, use the commands:

```
        awplus# configure terminal
awplus(config)# no aaa accounting dot1x default
```

Related Commands

aaa accounting update
aaa authentication dot1x
aaa group server
dot1x port-control
radius-server host

aaa accounting login

This command configures RADIUS and TACACS+ accounting for login shell sessions. The specified method list name can be used by the **accounting login** command in the Line Configuration mode. If the **default** parameter is specified, then this creates a default method list that is applied to every console and vty line, unless another accounting method list is applied on that line.

Note that unlimited RADIUS servers and up to four TACACS+ servers can be configured and consulted for accounting. The first server configured is regarded as the primary server and if the primary server fails then the backup servers are consulted in turn. A backup server is consulted if the primary server fails, i.e is unreachable.

Use the **no** variant of this command to remove an accounting method list for login shell sessions configured by an **aaa accounting login** command. If the method list being deleted is already applied to a console or vty line, accounting on that line will be disabled. If the default method list name is removed by this command, it will disable accounting on every line that has the default accounting configuration.

Syntax

```
aaa accounting login {default|<list-name>}
   {start-stop|stop-only|none} {group {radius|tacacs+|<group-name>}}

no aaa accounting login {default|<list-name>}
```

Parameter	Description
`default`	Default accounting method list.
`<list-name>`	Named accounting method list.
`start-stop`	Start and stop records to be sent.
`stop-only`	Stop records to be sent.
`none`	No accounting record to be sent.
`group`	Specify the servers or server group where accounting packets are sent.
`radius`	Use all RADIUS servers configured by the radius-server host command on page 44.6.
`tacacs+`	Use all TACACS+ servers configured by the tacacs-server host command.
`<group-name>`	Use the specified RADIUS server group, as configured by the aaa group server command.

Default Accounting for login shell sessions is disabled by default.

Mode Global Configuration

Usage This command enables you to define a named accounting method list. The items that you define in the accounting options are:

- the types of accounting packets that will be sent
- the set of servers to which the accounting packets will be sent

You can define a default method list with the name `default` and any number of other named method lists. The <*`list-name`*> for any method list that you define can then be used as the <*`list-name`*> parameter in the accounting login command available from Line Configuration mode.

If the method list name already exists, the command will replace the existing configuration with the new one.

There are two ways to define servers where RADIUS accounting messages are sent:

- **group radius**: use all RADIUS servers configured by radius-server host command
- **group <*group-name*>**: use the specified RADIUS server group configured with the aaa group server command

There is one way to define servers where TACACS+ accounting messages are sent:

- **group tacacs+**: use all TACACS+ servers configured by tacacs-server host command

The accounting event to send to the RADIUS or TACACS+ server is configured with the following options:

- **start-stop**: sends a **start** accounting message at the beginning of a session and a **stop** accounting message at the end of the session.
- **stop-only**: sends a **stop** accounting message at the end of a session.
- **none**: disables accounting.

Examples To configure RADIUS accounting for login shell sessions, use the following commands:

```
        awplus# configure terminal
awplus(config)# aaa accounting login default start-stop group
                radius
```

To configure TACACS+ accounting for login shell sessions, use the following commands:

```
        awplus# configure terminal
awplus(config)# aaa accounting login default start-stop group
                tacacs+
```

To reset the configuration of the default accounting list, use the following commands:

```
        awplus# configure terminal
awplus(config)#  no aaa accounting login default
```

Related Commands

aaa accounting commands
aaa authentication login
aaa accounting login
aaa accounting update
accounting login
radius-server host
tacacs-server host

aaa accounting update

This command enables periodic accounting reporting to either the RADIUS or TACACS+ accounting server(s) wherever login accounting has been configured.

Note that unlimited RADIUS servers and up to four TACACS+ servers can be configured and consulted for accounting. The first server configured is regarded as the primary server and if the primary server fails then the backup servers are consulted in turn. A backup server is consulted if the primary server fails, i.e is unreachable.

Use the **no** variant of this command to disable periodic accounting reporting to the accounting server(s).

Syntax

```
aaa accounting update [periodic <1-65535>]

no aaa accounting update
```

Parameter	Description
`periodic`	Send accounting records periodically.
<1-65535>	The interval to send accounting updates (in minutes). The default is 30 minutes.

Default Periodic accounting update is disabled by default.

Mode Global Configuration

Usage Use this command to enable the device to send periodic AAA login accounting reports to the accounting server. When periodic accounting report is enabled, interim accounting records are sent according to the interval specified by the **periodic** parameter. The accounting updates are start messages.

If the **no** variant of this command is used to disable periodic accounting reporting, any interval specified by the **periodic** parameter is reset to the default of 30 minutes when accounting reporting is reenabled, unless this interval is specified.

Example To configure the switch to send period accounting updates every 30 minutes, the default period, use the following commands:

```
        awplus# configure terminal
awplus(config)# aaa accounting update
```

To configure the switch to send period accounting updates every 10 minutes, use the following commands:

```
        awplus# configure terminal
awplus(config)# aaa accounting update periodic 10
```

To disable periodic accounting update wherever accounting has been configured, use the following commands:

```
        awplus# configure terminal
awplus(config)# no aaa accounting update
```

Related Commands aaa accounting auth-mac default
aaa accounting auth-web default
aaa accounting dot1x
aaa accounting login

aaa authentication auth-mac

This command enables MAC-based Port Authentication globally and allows you to specify an authentication method list. It is automatically applied to every interface running MAC-based Port Authentication.

Use the **no** variant of this command to globally disable MAC-based Port Authentication.

Syntax

```
aaa authentication auth-mac default group {<group-name>|radius}

no aaa authentication auth-mac default
```

Parameter	Description
<group-name>	Server group name.
radius	Use all RADIUS servers.

Default MAC-based Port Authentication is disabled by default.

Mode Global Configuration

Usage There are two ways to define servers where RADIUS accounting messages are sent:

- **group radius**: use all RADIUS servers configured by radius-server host command
- **group <*group-name*>**: use the specified RADIUS server group configured with the aaa group server command

All configured RADIUS Servers are automatically members of the server group **radius**. If a server is added to a named group **<*group-name*>**, it also remains a member of the group **radius**.

Example To enable MAC-based Port Authentication globally for all RADIUS servers, and use all available RADIUS servers, use the commands:

```
        awplus#  configure terminal
awplus(config)#  aaa authentication auth-mac default group
                 radius
```

To disable MAC-based Port Authentication, use the commands:

```
        awplus#  configure terminal
awplus(config)#  no aaa authentication auth-mac default
```

Related Commands aaa accounting auth-mac default
auth-mac enable

aaa authentication auth-web

This command enables Web-based Port Authentication globally and allows you to enable an authentication method list (in this case, a list of RADIUS Servers). It is automatically applied to every interface running Web-based Port Authentication.

Use the **no** variant of this command to globally disable Web-based Port Authentication.

Syntax

```
aaa authentication auth-web default group {<group-name>|radius}

no aaa authentication auth-web default
```

Parameter	Description
<group-name>	Server group name.
radius	Use all RADIUS servers.

Default Web-based Port Authentication is disabled by default.

Mode Global Configuration

Usage There are two ways to define servers where RADIUS accounting messages are sent:

- **group radius**: use all RADIUS servers configured by radius-server host command
- **group <*group-name*>**: use the specified RADIUS server group configured with the aaa group server command

Example To enable Web-based Port Authentication globally for all RADIUS servers, and use all available RADIUS servers, use the commands:

```
        awplus#  configure terminal
awplus(config)#  aaa authentication auth-web default group
                 radius
```

To disable Web-based Port Authentication, use the commands:

```
        awplus#  configure terminal
awplus(config)#  no aaa authentication auth-web default
```

Related Commands aaa accounting auth-web default
auth-mac enable

aaa authentication dot1x

This command enables 802.1X-based Port Authentication globally and allows you to enable an authentication method list. It is automatically applied to every interface running 802.1X-based Port Authentication.

Use the **no** variant of this command to globally disable 802.1X-based Port Authentication.

Syntax

```
aaa authentication dot1x default group {<group-name>|radius}

no aaa authentication dot1x default
```

Parameter	Description
radius	Use all RADIUS servers.
<*group-name*>	Server group name.

Default 802.1x-based Port Authentication is disabled by default.

Mode Global Configuration

Usage Use this command to specify the default method list to use for authentication on all switch ports with 802.1X enabled. Use the **no** variant of this command to reset the default authentication method list for 802.1X, to its default, that is, to use the group **radius**, containing all RADIUS servers configured by the **radius-server host** command.

There are two ways to define servers where RADIUS accounting messages are sent:

- **group radius**: use all RADIUS servers configured by radius-server host command
- **group <*group-name*>**: use the specified RADIUS server group configured with the aaa group server command

Example To enable 802.1X-based Port Authentication globally with all RADIUS servers, and use all available RADIUS servers, use the command:

```
        awplus# configure terminal
awplus(config)# aaa authentication dot1x default group radius
```

To disable 802.1X-based Port Authentication, use the command:

```
        awplus# configure terminal
awplus(config)# no aaa authentication dot1x default
```

Related Commands

aaa accounting dot1x
aaa group server
dot1x port-control
radius-server host

aaa authentication enable default group tacacs+

This command enables AAA authentication to determine the privilege level a user can access for passwords authenticated against the TACACS+ server.

Use the **no** variant of this command to disable privilege level authentication.

Syntax

```
aaa authentication enable default group tacacs+ [local] [none]

no aaa authentication enable default
```

Parameter	Description
`local`	Use the locally configured enable password (**enable password** command) for authentication.
`none`	No authentication.

Default Local privilege level authentication is enabled by default (aaa authentication enable default local command).

Mode Global Configuration

Usage A user is configured on a TACACS+ server with a maximum privilege level. When they enter the enable (Privileged Exec mode) command they are prompted for an enable password which is authenticated against the TACACS+ server. If the password is correct and the specified privilege level is equal to or less than the users maximum privilege level, then they are granted access to that level. If the user attempts to access a privilege level that is higher than their maximum configured privilege level, then the authentication session will fail and they will remain at their current privilege level.

Note If both **local** and **none** are specified, you must always specify **local** first.

If the TACACS+ server goes offline, or is not reachable during enable password authentication, and command level authentication is configured as:

- **aaa authentication enable default group tacacs+**
  then the user is never granted access to Privileged Exec mode.
- **aaa authentication enable default group tacacs+ local**
  then the user is authenticated using the locally configured enable password, which if entered correctly grants the user access to Privileged Exec mode. If no enable password is locally configured (**enable password** command), then the enable authentication will fail until the TACACS+ server becomes available again.
- **aaa authentication enable default group tacacs+ none**
  then the user is granted access to Privileged Exec mode with no authentication. This is true even if a locally configured enable password is configured.
- **aaa authentication enable default group tacacs+ local none**
  then the user is authenticated using the locally configured enable password. If no enable password is locally configured, then the enable authentication will grant access to Privileged Exec mode with no authentication.

If the password for the user is not successfully authenticated by the server, then the user is again prompted for an enable password when they enter **enable** via the CLI.

Example To enable a privilege level authentication method that will not allow the user to access Privileged Exec mode if the TACACS+ server goes offline, or is not reachable during enable password authentication, use the following commands:

```
        awplus# configure terminal
awplus(config)# aaa authentication enable default group tacacs+
```

To enable a privilege level authentication method that will allow the user to access Privileged Exec mode if the TACACS+ server goes offline, or is not reachable during enable password authentication, and a locally configured enable password is configured, use the following commands:

```
        awplus# configure terminal
awplus(config)# aaa authentication enable default group tacacs+
                local
```

To disable privilege level authentication, use the following commands:

```
        awplus# configure terminal
awplus(config)# no aaa authentication enable default
```

Related Commands aaa authentication login
aaa authentication enable default local
enable (Privileged Exec mode)
enable password
enable secret
tacacs-server host

aaa authentication enable default local

This command enables AAA authentication to determine the privilege level a user can access for passwords authenticated locally.

Syntax `aaa authentication enable default local`

Default Local privilege level authentication is enabled by default.

Mode Global Configuration

Usage The privilege level configured for a particular user in the local user database is the privilege threshold above which the user is prompted for an enable (Privileged Exec mode) command.

Example To enable local privilege level authentication command, use the following commands:

```
        awplus# configure terminal
awplus(config)# aaa authentication enable default local
```

To disable privilege level authentication, use the following commands:

```
        awplus# configure terminal
awplus(config)# no aaa authentication enable default
```

Related Commands

aaa authentication enable default group tacacs+
aaa authentication login
enable (Privileged Exec mode)
enable password
enable secret
tacacs-server host

aaa authentication login

Use this command to create an ordered list of methods to use to authenticate user login, or to replace an existing method list with the same name. Specify one or more of the options **local** or **group**, in the order you want them to be applied. If the **default** method list name is specified, it is applied to every console and VTY line immediately unless another method list is applied to that line by the login authentication command. To apply a non-default method list, you must also use the login authentication command.

Use the **no** variant of this command to remove an authentication method list for user login. The specified method list name is deleted from the configuration. If the method list name has been applied to any console or VTY line, user login authentication on that line will fail.

Note that the **no aaa authentication login default** command does not remove the default method list. This will return the default method list to its default state (**local** is the default).

Syntax

```
aaa authentication login {default|<list-name>}
   {[local] [group {radius|tacacs+|<group-name>}]}
```

```
no aaa authentication login {default|<list-name>}
```

Parameter	Description
`default`	Set the default authentication server for user login.
`<list-name>`	Name of authentication server.
`local`	Use the local username database.
`group`	Use server group.
`radius`	Use all RADIUS servers configured by the radius-server host command on page 44.6.
`tacacs+`	Use all TACACS+ servers configured by the tacacs-server host command.
`<group-name>`	Use the specified RADIUS server group, as configured by the aaa group server command.

Default If the default server is not configured using this command, user login authentication uses the local user database only.

If the **default** method list name is specified, it is applied to every console and VTY line immediately unless a named method list server is applied to that line by the **login authentication** command.

local is the default state for the default method list unless a named method list is applied to that line by the **login authentication** command. Reset to the default method list using the **no aaa authentication login default** command.

Mode Global Configuration

Usage When a user attempts to log in, the switch sends an authentication request to the first authentication server in the method list. If the first server in the list is reachable and it contains a username and password matching the authentication request, the user is authenticated and the login succeeds. If the authentication server denies the authentication request because of an incorrect username or password, the user login fails. If the first server in the method list is unreachable, the switch sends the request to the next server in the list, and so on.

For example, if the method list specifies group tacacs+ local, and a user attempts to log in with a password that does not match a user entry in the first TACACS+ server, if this TACACS+

server denies the authentication request, then the switch does not try any other TACACS+ servers not the local user database; the user login fails.

Examples To configure the default authentication method list for user login to use first all available RADIUS servers for user login authentication and then the local user database, use the following commands:

```
awplus# configure terminal
awplus(config)# aaa authentication login default group radius
                local
```

To configure a user login authentication method list called USERS to use first the RADIUS server group RAD_GROUP1 for user login authentication and then the local user database, use the following commands:

```
awplus# configure terminal
awplus(config)# aaa authentication login USERS group
                RAD_GROUP1 local
```

To configure a user login authentication method list called USERS to use first the TACACS+ servers for user login authentication and then the local user database, use the following commands:

```
awplus# configure terminal
awplus(config)# aaa authentication login USERS group tacacs+
                local
```

To return to the default method list (**local** is the default server), use the following commands:

```
awplus# configure terminal
awplus(config)# no aaa authentication login default
```

To delete an existing authentication method list USERS created for user login authentication, use the following commands:

```
awplus# configure terminal
awplus(config)# no aaa authentication login USERS
```

Related Commands aaa accounting commands
aaa authentication enable default group tacacs+
login authentication

aaa group server

This command configures a RADIUS server group. A server group can be used to specify a subset of RADIUS servers in **aaa** commands. The group name **radius** is predefined, which includes all RADIUS servers configured by the **radius-server host** command.

RADIUS servers are added to a server group using the **server** command. Each RADIUS server should be configured using the **radius-server host** command.

Use the **no** variant of this command to remove an existing RADIUS server group.

Syntax

```
aaa group server radius <group-name>

no aaa group server radius <group-name>
```

Parameter	Description
<group-name>	Server group name.

Mode Global Configuration

Usage Use this command to create an AAA group of RADIUS servers, and to enter Server Group Configuration mode, in which you can add servers to the group. Use a server group to specify a subset of RADIUS servers in AAA commands. Each RADIUS server must be configured by the **radius-server host** command. To add RADIUS servers to a server group, use the **server** command.

Example To create a RADIUS server group named GROUP1 with hosts 192.168.1.1, 192.168.2.1 and 192.168.3.1, use the commands:

```
   awplus(config)# aaa group server radius GROUP1

awplus(config-sg)# server 192.168.1.1 auth-port 1812 acct-
                   port 1813

awplus(config-sg)# server 192.168.2.1 auth-port 1812 acct-
                   port 1813

awplus(config-sg)# server 192.168.3.1 auth-port 1812 acct-
                   port 1813
```

To remove a RADIUS server group named GROUP1 from the configuration, use the command:

```
awplus(config)# no aaa group server radius GROUP1
```

Related Commands

aaa accounting auth-mac default
aaa accounting auth-web default
aaa accounting dot1x
aaa accounting login
aaa authentication auth-mac
aaa authentication auth-web
aaa authentication dot1x
aaa authentication login
radius-server host
server (Server Group)

aaa local authentication attempts lockout-time

This command configures the duration of the user lockout period.

Use the **no** variant of this command to restore the duration of the user lockout period to its default of 300 seconds (5 minutes).

Syntax

```
aaa local authentication attempts lockout-time <lockout-tiime>

no aaa local authentication attempts lockout-time
```

Parameter	Description
<lockout-time>	<0-10000>. Time in seconds to lockout the user.

Mode Global Configuration

Default The default for the lockout-time is 300 seconds (5 minutes).

Usage While locked out all attempts to login with the locked account will fail. The lockout can be manually cleared by another privileged account using the clear aaa local user lockout command.

Example To configure the lockout period to 10 minutes (600 seconds), use the commands:

```
        awplus# configure terminal
awplus(config)# aaa local authentication attempts lockout-time
                600
```

To restore the default lockout period of 5 minutes (300 seconds), use the commands:

```
        awplus# configure terminal
awplus(config)# no aaa local authentication attempts
                lockout-time
```

Related Commands aaa local authentication attempts max-fail

aaa local authentication attempts max-fail

This command configures the maximum number of failed login attempts before a user account is locked out. Every time a login attempt fails the failed login counter is incremented.

Use the **no** variant of this command to restore the maximum number of failed login attempts to the default setting (5 failed login attempts).

Syntax

```
aaa local authentication attempts max-fail <failed-logins>

no aaa local authentication attempts max-fail
```

Parameter	Description
<failed-logins>	<1-32>. Number of login failures allowed before locking out a user.

Mode Global Configuration

Default The default for the maximum number of failed login attempts is 5 failed login attempts.

Usage When the failed login counter reaches the limit configured by this command that user account is locked out for a specified duration configured by the aaa local authentication attempts lockout-time command.

When a successful login occurs the failed login counter is reset to 0. When a user account is locked out all attempts to login using that user account will fail.

Example To configure the number of login failures that will lock out a user account to 2 login attempts, use the commands:

```
        awplus# configure terminal
awplus(config)# aaa local authentication attempts max-fail 2
```

To restore the number of login failures that will lock out a user account to the default number of login attempts (5 login attempts), use the commands:

```
        awplus# configure terminal
awplus(config)# no aaa local authentication attempts max-fail
```

Related Commands aaa local authentication attempts lockout-time
clear aaa local user lockout

accounting login

This command applies a login accounting method list to console or vty lines for user login. When login accounting is enabled using the **aaa accounting login** command, logging events generate an accounting record to the accounting server configured using **aaa accounting login**.

The accounting method list must be configured first using the **aaa accounting login** command. If an accounting method list is specified that has not been created by the **aaa accounting login** command then accounting will be disabled on the specified lines.

The **no** variant of this command resets AAA (Authentication, Authorization, Accounting) Accounting applied to console or vty lines for local or remote login. **default** login accounting is applied after issuing the **no accounting login** command. Accounting is disabled with **default**.

Syntax

```
accounting login {default|<list-name>}
```

```
no accounting login
```

Parameter	Description
`default`	Default accounting method list.
`<list-name>`	Named accounting method list.

Default By default login accounting is disabled in the **default** accounting server. No accounting will be performed until accounting is enabled using the **aaa accounting login** command beforehand.

Mode Line Configuration

Example To apply the accounting server `USERS` to all vty lines use the following commands:

```
awplus# configure terminal
awplus(config)# line vty 0 32
awplus(config-line)# accounting login USERS
```

To reset accounting for login sessions on the console, use the following commands:

```
awplus# configure terminal
awplus(config)# line console 0
awplus(config-line)# no accounting login
```

Related Commands aaa accounting commands
aaa accounting login

clear aaa local user lockout

Use this command to clear the lockout on a specific user account or all user accounts.

Syntax

```
clear aaa local user lockout {username <username>|all}
```

Parameter	Description
`username`	Clear lockout for the specified user.
`<username>`	Specifies the user account.
`all`	Clear lockout for all user accounts.

Mode Privileged Exec

Examples To unlock the user account 'bob' use the following command:

```
awplus# clear aaa local user lockout username bob
```

To unlock all user accounts use the following command:

```
awplus# clear aaa local user lockout all
```

Related Commands aaa local authentication attempts lockout-time

debug aaa

This command enables AAA debugging.

Use the **no** variant of this command to disable AAA debugging.

Syntax

```
debug aaa [accounting|all|authentication|authorization]

no debug aaa [accounting|all|authentication|authorization]
```

Parameter	Description
`accounting`	Accounting debugging.
`all`	All debugging options are enabled.
`authentication`	Authentication debugging.
`authorization`	Authorization debugging.

Default AAA debugging is disabled by default.

Mode Privileged Exec

Example To enable authentication debugging for AAA, use the command:

```
awplus# debug aaa authentication
```

To disable authentication debugging for AAA, use the command:

```
awplus# no debug aaa authentication
```

Related Commands show debugging aaa
undebug aaa

login authentication

Use this command to apply an AAA server for authenticating user login attempts from a console or remote logins on these console or VTY lines. The authentication method list must be specified by the **aaa authentication login** command. If the method list has not been configured by the **aaa authentication login** command, login authentication will fail on these lines.

Use the **no** variant of this command to reset AAA Authentication configuration to use the default method list for login authentication on these console or VTY lines.

Command Syntax

```
login authentication {default|<list-name>}

no login authentication
```

Parameter	Description
`default`	The default authentication method list. If the default method list has not been configured by the aaa authentication login command, the local user database is used for user login authentication.
`<list-name>`	Named authentication server.

Default The default login authentication method list, as specified by the aaa authentication login command, is used to authenticate user login. (If this has not been specified, the default is to use the local user database.)

Mode Line Configuration

Examples To apply the authentication method list called `CONSOLE` to the console port terminal line (asyn 0), use the following commands:

```
           awplus# configure terminal
   awplus(config)# line console 0
awplus(config-line)# login authentication CONSOLE
```

To reset user authentication configuration on all VTY lines, use the following commands:

```
           awplus# configure terminal
   awplus(config)# line vty 0 32
awplus(config-line)# no login authentication
```

Related Commands aaa authentication login
line

show debugging aaa

This command displays the current debugging status for AAA (Authentication, Authorization, Accounting).

Syntax

```
show debugging aaa
```

Mode User Exec and Privileged Exec

Example To display the current debugging status of AAA, use the command:

```
awplus# show debug aaa
```

Output Figure 42-1: Example output from the **show debug aaa** command

```
AAA debugging status:
    Authentication debugging is on
    Accounting debugging is off
```

undebug aaa

This command applies the functionality of the no debug aaa command on page 42.27.

Chapter 43: RADIUS Introduction and Configuration

Introduction

The main purpose of RADIUS (Remote Authentication Dial In User Service) is to enable the authentication of network users stored in a database on a server known as a RADIUS Server.

When users connect to the network, the switch the users connect to can challenge the users for authentication, and pass on the authentication to the RADIUS server to check. Based on the result of the check against the database, the RADIUS Server informs the switch whether or not to allow the connected user access to the network.

A RADIUS Server can do more than allow or deny access to the network. A RADIUS Server can send back parameters to the connected users, such as an IP address for the user, or a VLAN for the user, or a privilege level for a session. RADIUS also provides an accounting service. Switches can inform the RADIUS Server how long a user has been connected to the network, and how much traffic the user has sent and received while connected to the network.

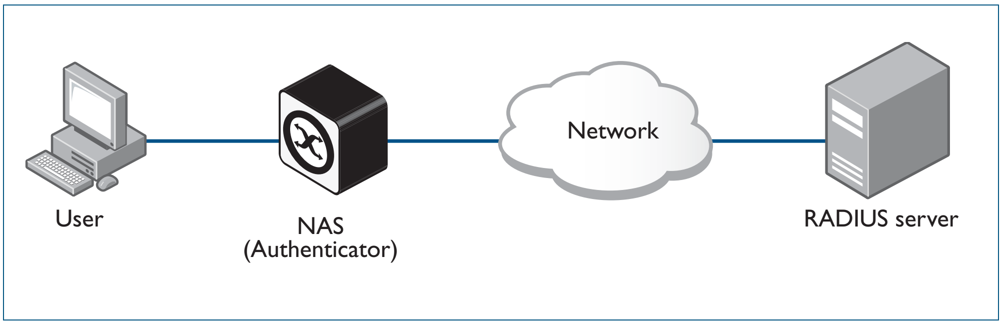
The original use for RADIUS was for the authentication of users dialling into an ISP (Internet Service Provider). A PPP (Point-to-Point Protocol) connection would be established between the remote client and the ISP's access switch. The ISP's access switch would receive the client's username and password using PAP (Password Authentication Protocol) or using CHAP (Challenge Handshake Authentication Protocol) and pass on the client's username and password to the RADIUS server to authenticate the client. The RADIUS Server's response to the authentication request would be sent back to the client as a PAP or CHAP allow or deny.

RADIUS has been adapted to network access authentication applications. Network access authentication using RADIUS follows a similar method to the PPP dial-up application for ISPs. For general network access authentication there is the RADIUS Server where the database of user authentication data is stored and a NAS (Network Access Server), which is the switch that user connects to first. The RADIUS Server and the NAS communicate with each other through exchanging attributes. Usernames and passwords are treated as attributes in RADIUS packets to and from a RADIUS Server and a NAS. The RADIUS Server is configured with a list of valid NASs that are allowed to send authentication requests to the RADIUS Server.

The RADIUS Server will not accept authentication requests from a NAS that is not on the list of valid NASs. Each NAS has a shared secret, which is a shared key with the RADIUS Server that is used to authenticate requests. The RADIUS Server has access to a list of user authentication data, stored within the RADIUS Server or accessed from another server.

Communication between the NAS and RADIUS Server uses the RADIUS protocol. The RADIUS protocol uses UDP packets. There are two UDP ports used as the destination port for RADIUS authentication packets (ports 1645 and 1812). Note that port 1812 is in more common use than port 1645 for authentication packets. UDP ports (ports 1646 and 1813) are used for RADIUS accounting separately from the ports used for RADIUS authentication.

Figure 43-1: Example showing a User to a NAS to a RADIUS Server network connection



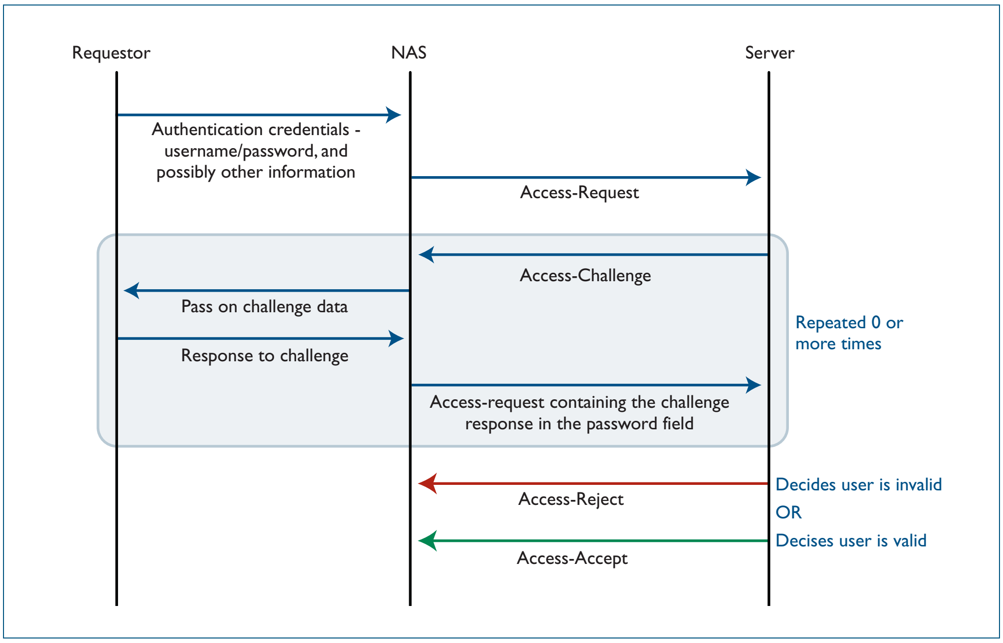
RADIUS Packets

The RADIUS RFCs define the RADIUS packet types and attributes. RADIUS authentication is defined by RFC2058, RFC2138, RFC2865, and RFC2868. RADIUS accounting is defined by RFC2059, RFC2139, RFC2866, and RFC2867. These RADIUS RFCs define over fifty attributes and six packets types (`Access-Request, Access-Accept, Access-Reject, Accounting-Request, Accounting-Response, Access-Challenge`).

A RADIUS exchange is initiated by the NAS when a user requests access to the NAS. The NAS obtains the user authentication data adds them into a RADIUS `Access-Request` packet type and sends the RADIUS `Access-Request` packet to the RADIUS Server.

- If a RADIUS Server has not been configured for authentication request from a NAS then it will silently discard an `Access-Request` packet from it.
- If the RADIUS Server accepts the request from the NAS it considers the authentication date provided in the `Access-Request` packet. The RADIUS Server may verify the user from its own database or it may connect to other servers to verify.
- If the RADIUS Server decides that the user is not allowed access to the NAS it responds to the NAS with an `Access-Reject` packet and the NAS will block the user.
- If the RADIUS Server decides that the user is valid but needs more information to verify that the user is not an imposter, it may send an `Access-Challenge` packet to the NAS that the NAS forwards to the user. The NAS forwards the user response to the `Access-Challenge` packet in an `Access-Request` packet to the RADIUS Server to accept or reject to allow or deny NAS user access.
- If the RADIUS Server rejects the user it sends an `Access-Reject` packet to the NAS.
- If the RADIUS Server accepts the user it sends an `Accept-Accept` packet to the NAS. The `Accept-Accept` packet to the NAS contains attributes that the NAS can apply.

Figure 43-2: Example showing an exchange from a Requestor to a NAS to a RADIUS Server



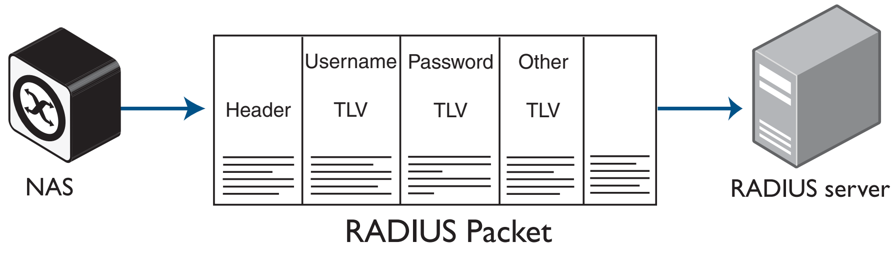
RADIUS Attributes

Each attribute is identified by its RFC-defined name, followed by its attribute ID in parenthesis.

- **User-name(1)**
  User-names are strings of at least three characters and have a maximum of 253 characters, which is the upper limit on all RADIUS attributes.
- **User-password(2)**
  User-passwords are encrypted using an MD5 hash of the password, the NAS's shared secret with the RADIUS Server, and a request authenticator value. User-passwords can either be used at the initial authentication attempt or in response to an Access-Challenge packet type from the RADIUS Server to the NAS.
- **CHAP-password(3)**
  CHAP-passwords are used if the NAS is using CHAP to authenticate the user, and doesn't receive the use the user's password but sends the CHAP response to the RADIUS Server instead. The CHAP password is an encrypted string that is an MD5 hash of the password and challenge value sent by the user.
- **Framed-IP-Address(8)**
  Used for dial-in user making PPP connections to the NAS who are dynamically allocated an IP address that they can use for the duration of their connect. The RADIUS Server sends the Framed-IP-Address to the NAS to allocate.
- **Service-Type(6)**
  Used when the NAS is authenticating a user who wants to open a management session on the NAS, and is sent by the RADIUS Server back to the NAS in an Access-Accept type packet to indicate the level of access the NAS gives a user. Service-Type(6) is mapped to a Privileged management session for AlliedWare Plus.
- **NAS-Port-Type(61)**
  Identifies the type of port on which the user is accessing the NAS. The NAS-Port-Type(61) attribute is sent by the NAS to the RADIUS Server in Access-Request type packet, so the RADIUS Server may use it to choose access type. For 802.1X sessions, the NAS-Port-Type sent by the NAS is Ethernet (15).
- **802.1X VLAN assignment uses:**
  Tunnel-Type(64), Tunnel-Medium-Type(65), Tunnel-Private-Group-ID(81), Egress-VLANID(56), and Egress-VLAN-Name(58) attributes (specified in RFC4675 used to specify 802.1Q tagged and untagged VLAN assignments with LLDP-MED/Voice-VLAN).

Attributes are carried within RADIUS packets in the form of TLVs (Type Length Values). Every attribute has an attribute ID number in the Type field of the TLV. The Length field holds a one-byte number that represents then length of the TLV. The Value field holds the value of the attribute.

Figure 43-3: Example showing **TLVs** in a RADIUS Packet from a NAS to a RADIUS Server



RADIUS Security

RADIUS is used for network security and carries user authentication information, so can be a target for security attacks. To counter threats there are three elements to RADIUS security:

- Shared secret
- Authenticator
- Password Encryption

Shared Secret

Every NAS and server are configured with a pre-shared key, called the "shared secret", which is a key string, with no particular format of at least 16 characters.

The protocol has no method for choosing and sharing the secret between the NAS and the server. The secret must be manually generated and separately configured on the NAS and on the server.

The shared secret itself never appears in any RADIUS packets. It is used as an input to the algorithms used for creating encrypted values that are carried in the packets.

Authenticator

The authenticator is a random 16-byte value generated by the NAS. The NAS creates a new authenticator value for each `Access-Request` that it sends.

The response packets that come back from the server contain a value called the Response Authenticator. This is a value that is created by performing an MD5 hash on a string that is created by concatenating the packet type identifier, Session ID, Authenticator sent in the request packet, Attribute fields in the packet, Shared secret that the server shares with the NAS to which it is responding.

When the NAS receives the response packet, it performs the same hash on the same values, and verifies that it comes up with the same result. If not, then it must assume that the response packet has been spoofed, and silently discards it.

Password Encryption

The value placed in the user-password TLV of an `Access-Request` packet is not simply an exact copy of the password sent from the requestor to the NAS.

The NAS concatenates together the shared secret and the authenticator that it has randomly generated for this request and then performs manipulations (MD5, XOR) on that concatenation, and the password to create the value to go into password TLV.

When the server validates the `Access-Request`, it retrieves the user's password from the user credentials database, and performs the same manipulation upon that password. If the result matches the value in the user-password field of the `Access-Request`, then the password sent by the requestor is deemed to be correct.

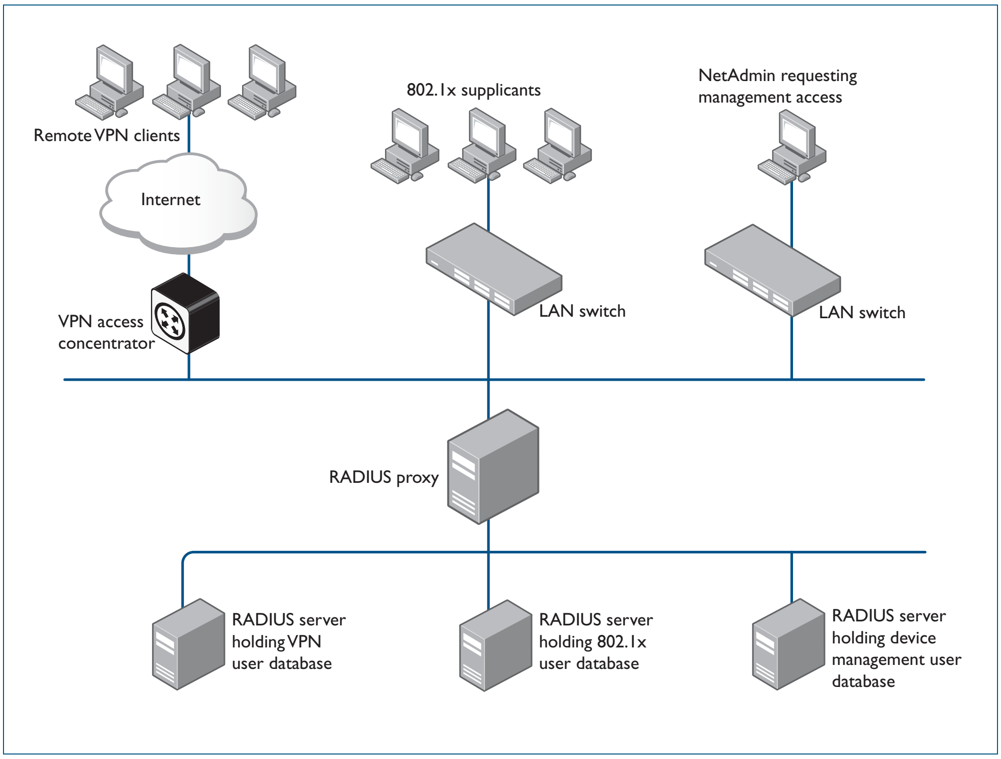
RADIUS Proxy

The user database, which user credentials sent to a RADIUS server are looked up in, may not reside on the RADIUS server itself. The external user database may reside on another RADIUS server, and the communication to that server uses RADIUS. In the case where a RADIUS server communicates with a NAS, but also acts as a client to another RADIUS server, is said to be acting as a RADIUS proxy.

There are a variety of situations where RADIUS proxy is useful. Multiple RADIUS servers could have been set up, holding user databases for different purposes such as Authentication, Switch management sessions, Authenticating VPN connections, and Authenticating 802.1X sessions.

But it is convenient for there to be just one address that all the NASs in the network use as their RADIUS server. That one RADIUS server that the NASs send their requests to, can act as a proxy for all the servers holding the different user databases.

Figure 43-4: Example showing RADIUS Proxy


Remote VPN clients
Internet
VPN access concentrator
802.1x supplicants
LAN switch
NetAdmin requesting management access
LAN switch
RADIUS proxy
RADIUS server holding VPN user database
RADIUS server holding 802.1x user database
RADIUS server holding device management user database

RADIUS Accounting

There are only two types of RADIUS accounting packet: `Accounting-Request` and `Accounting-Response`.

The `Accounting-Request` packets are always sent from the NAS to the server. The `Accounting-Response` packets are always sent from the server to the NAS, and are effectively ACKs of the `Accounting-Request` packets.

The `Accounting-Request` packets always carry the attribute `Acct-Status-Type`. The most commonly used values of this attribute are:

- **Start** – which denotes a packet marking that a session is beginning
- **Stop** – which denotes a packet marking that a session is ending
- **Interim update** – packets sent periodically during the session to give update reports on the statistics that are being collected.

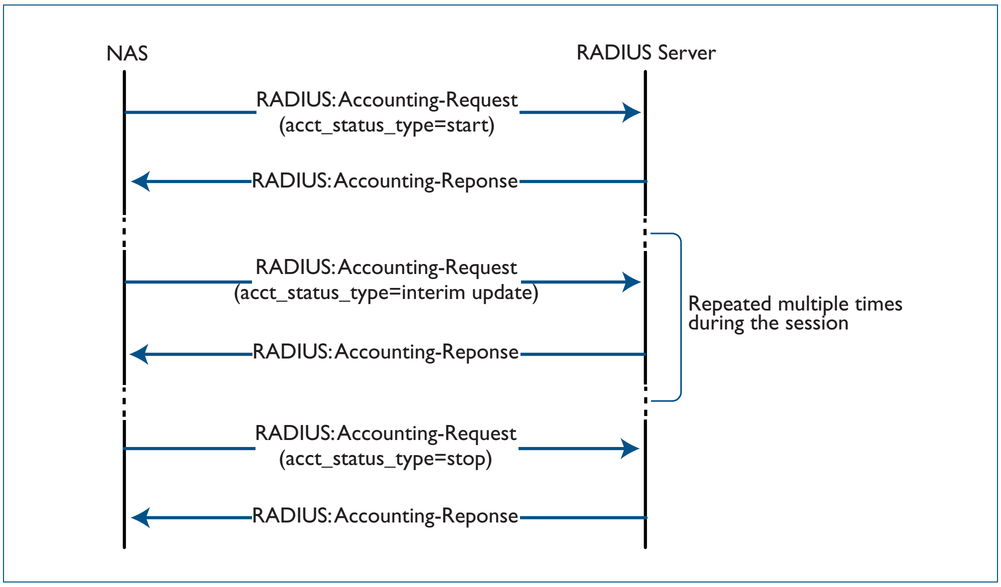
The statistics that can be exchanged in the session are:

- Input Octets
- Input Packets
- Output Octets
- Output Packets
- Session Duration

There is no requirement to exchange all these statistics – NAS implementations are at liberty to choose which statistics they will send. Each of these statistics has a corresponding attribute type. The attributes are sent in Interim-Update and Stop accounting request packets.

Each accounting session has a unique session ID, which is chosen by the NAS. The session ID is carried in an `Acct-Session-Id` attribute, that should be present in every packet involved in the session. The accounting packets typically do not use the same UDP port as the authentication packets. The default port for RADIUS accounting is 1813.

Figure 43-5: Example showing RADIUS Accounting between a NAS and a RADIUS Server


NAS
RADIUS Server
RADIUS:Accounting-Request
(acct_status_type=start)
RADIUS:Accounting-Reponse
RADIUS:Accounting-Request
(acct_status_type=interim update)
RADIUS:Accounting-Reponse
Repeated multiple times during the session
RADIUS:Accounting-Request
(acct_status_type=stop)
RADIUS:Accounting-Reponse

RADIUS Configuration

This section describes how to configure RADIUS with the available AAA commands. For a description of AAA commands, refer to the AAA Commands chapter. For a description of the RADIUS commands used, refer to the RADIUS Commands chapter.

RADIUS is often used in a variety of networks that need high security while maintaining access for remote users. RADIUS is suitable for the following networks that require access security:

- Networks with multiple-vendor access servers, each supporting RADIUS. For example, access servers from several vendors use a single RADIUS server-based security database.
- Networks in which a user may access a single service. Using RADIUS, you can control user access to a single host, or to a single utility such as Telnet.
- Networks that require accounting. You can use RADIUS accounting independent of RADIUS authentication. The RADIUS accounting functions allow data to be sent at the start and end of services, indicating the amount of resources (time, packets, bytes) used.

Switch Configuration Tasks

To configure RADIUS on your switch or access server, you must perform the following tasks:

- Use the **aaa authentication** command to define method lists for RADIUS authentication. For information about this command, refer to the AAA Commands chapter.
- Use authentication commands to enable the defined method lists to be used. For more information, refer to the Authentication Commands chapter.

The following configuration tasks are optional:

- You can use the aaa group server command to group selected RADIUS hosts for specific services. For detailed information about this command, refer to the AAA Server Groups Configuration section in this chapter and refer to the AAA Commands chapter.
- You can use the aaa accounting login command to enable accounting for RADIUS connections. For information about this command, refer to the AAA Commands chapter.

This section describes how to set up RADIUS for authentication and accounting on your network, and includes the following sections:

- Switch to RADIUS Server Communication (Required)
- Configuring AAA Server Groups (Optional)
- Configuring AAA Server Groups with Deadtime (Optional)
- Specifying RADIUS Authentication
- Specifying RADIUS Accounting (Optional)

For RADIUS configuration examples using the commands in this chapter, refer to the section RADIUS Configuration Examples at the end of this chapter.

Switch to RADIUS Server Communication

The RADIUS host is normally a multiuser system running RADIUS server software from a software provider. Switch to RADIUS server communication has several components:

- Host name or IP address
- Authentication destination port
- Accounting destination port
- Timeout period
- Retransmission value
- Key string

RADIUS security servers are identified on the basis of their host name or IP address, host name and specific UDP port numbers, or IP address and specific UDP port numbers. The combination of the IP address and UDP port number creates a unique identifier, allowing different ports to be individually defined as RADIUS hosts providing a specific AAA service. This unique identifier enables RADIUS requests to be sent to multiple UDP ports on a server at the same IP address.

A RADIUS server and a switch use a shared secret text string to encrypt passwords and exchange responses. To configure RADIUS using the AAA security commands, you must specify the host running the RADIUS server daemon and a secret text string that it shares with the switch, which you can specify using the **key** parameter in the radius-server host command.

The timeout, retransmission, and encryption key values are configurable globally for all RADIUS servers, on a per-server basis, or in some combination of global and per-server settings. To apply these settings globally to all RADIUS servers communicating with the switch, use the three global commands: radius-server timeout, radius-server retransmit, and radius-server key. To apply these values on a specific RADIUS server, use the radius-server host command.

Note You can configure both global and per-server timeout, retransmission, and key value commands simultaneously on the same Network Access Server.

If both global and per-server functions are configured on a switch, the per-server timer, retransmission, and key value commands override global timer, retransmission, and key value commands.

To configure per-server RADIUS server communication, use the following command in the Global Configuration mode:

Mode and Command	Command Purpose

`awplus(config)#`
`radius-server host {<hostname>| <ip-address>} [auth-port <port-number>] [acct-port <port-number>] [timeout <seconds>] [retransmit <retries>] [key <string>]`
	Specifies the IP address or host name of the remote RADIUS server host and assigns authentication and accounting destination UDP port numbers.
	Use the `auth-port <port-number>` option to configure a specific UDP port on this RADIUS server to be used solely for authentication.
	Use the `acct-port <port-number>` option to configure a specific UDP port on this RADIUS server to be used solely for accounting.
	To configure the network access server to recognize more than one host entry associated with a single IP address, simply repeat this command as many times as necessary, making sure that each UDP port number is different.
	Set the timeout, retransmit, and encryption key values to use with the specific RADIUS host. If no timeout is set, the global value is used; otherwise, enter a value in the range 1 to 1000.
	If no retransmit value is set, the global value is used; otherwise enter a value in the range 1 to 1000. If no key string is specified, the global value is used.

To configure global communication settings between the switch and a RADIUS server, use the following **radius-server** commands in the Global Configuration mode:

Mode and Command	Command Purpose

`awplus(config)#`
`radius-server key <key>`
	Specifies the shared secret text string used between the switch and a RADIUS server (no default is set).

`awplus(config)#`
`radius-server retransmit <retries>`
	Specifies how many times the switch transmits each RADIUS request to the RADIUS server before giving up (the default is 3).

`awplus(config)#`
`radius-server timeout <seconds>`
	Specifies for how many seconds a switch waits for a reply to a RADIUS request before retransmitting the request.

`awplus(config)#`
`radius-server deadtime <minutes>`
	Specifies for how many minutes a RADIUS server that is not responding to authentication requests is passed over by requests for RADIUS authentication.

AAA Server Groups Configuration

Configuring the switch to use AAA server groups provides a way to group existing server hosts. This allows you to select a subset of the configured server hosts and use them for a particular service. A server group is used in conjunction with a global server-host list. The server group lists the IP addresses of the selected server hosts.

Server groups also can include multiple host entries for the same server, as long as each entry has a unique identifier. The combination of an IP address and a UDP port number creates a unique identifier, allowing different ports to be individually defined as RADIUS hosts providing a specific AAA service.

To define a server host with a server group name, enter the following commands in the Global Configuration mode. The listed RADIUS server must exist in the Global Configuration mode:

Mode and Command	Command Purpose
awplus(config)# `radius-server host {<hostname>|<ip-address>} [auth-port <port-number>] [acct-port <port-number>] [timeout <seconds>] [retransmit <retries>] [key <string>]`	Specifies and defines the IP address of the server host before configuring the AAA server-group. Refer to the section Switch to RADIUS Server Communication of this chapter for more information on the radius-server host command.
awplus(config-if)# `aaa group server <group-name>`	Defines the AAA server group with a group name. This command puts the switch in server group sub configuration mode.
awplus(config-sg)# `server {<hostname>|<ip-address>} [auth-port <port-number>] [acct-port <port-number>]`	Associates a particular RADIUS server with the defined server group. Each security server is identified by its IP address and UDP port number. Repeat this step for each RADIUS server in the AAA server group. Each server in the group must be defined previously using the radius-server host command.

Configuring AAA Server Groups with Deadtime

After you have configured a server host with a server name, you can use the deadtime (RADIUS server group) command to configure each server per server group. Configuring deadtime within a server group allows you to direct AAA traffic to separate groups of servers that have different operational characteristics.

Configuring **deadtime** is no longer limited to a global configuration. A separate timer has been attached to each server host in every server group. When a server is found to be unresponsive after numerous retransmissions and time-outs, the server is assumed to be dead. The timers attached to each server host in all server groups are triggered. In essence, the timers are checked and subsequent requests to a server, once it is assumed to be dead, are directed to alternate servers, if configured. When the network access server receives a reply from the server, it checks and stops all configured timers, if running, for that server in all server groups.

If the timer has expired, only the server to which the timer is attached is assumed to be alive. This becomes the only server that can be tried for later AAA requests using the server groups to which the timer belongs.

Note Since one server has different timers and may have different deadtime values configured in the server groups, the same server may in the future have different states, dead and alive, at the same time. To change the state of a server, you must start and stop all configured timers in all server groups.

The size of the server group will be increased because of the addition of new timers and the deadtime attribute. The overall impact of the structure depends on the number and size of the server groups and how the servers are shared among server groups in a specific configuration.

To configure deadtime within a server group, use the following commands beginning in the Global Configuration mode:

Mode and Command	Command Purpose

`awplus(config)#`
`aaa group server radius group1`
	Defines a RADIUS type server group.

`awplus(config-sg)#`
`deadtime 1`
	Configures and defines a deadtime value in minutes.

`awplus(config-sg)#`
`exit`
	Exits server group configuration mode.

Specifying RADIUS Authentication

After you have identified the RADIUS server and defined the RADIUS authentication key, you must define method lists for RADIUS authentication. Because RADIUS authentication is facilitated through AAA, you must enter the aaa authentication login command, specifying RADIUS as the authentication method. For detailed aaa authentication login command information, refer to the AAA Commands chapter.

Specifying RADIUS Accounting

The AAA accounting feature enables you to track the services users are accessing as well as the amount of network resources they are consuming. Because RADIUS accounting is facilitated through AAA, you must issue the aaa accounting login command, specifying RADIUS as the accounting method. For detailed aaa accounting login command information, refer to the AAA Commands chapter.

Monitoring and Maintaining RADIUS

To monitor and maintain RADIUS, use the following commands in Privileged Exec mode:

Mode and Command	Command Purpose

`awplus#`
`debug radius`
	Displays information associated with RADIUS.
	For detailed debug radius command information, refer to the RADIUS Commands chapter.

`awplus#`
`show radius statistics`
	Displays the RADIUS statistics for accounting and authentication packets.
	For detailed show radius statistics command information, refer to the RADIUS Commands chapter.

RADIUS Configuration Examples

The following sections provide RADIUS configuration examples:

- RADIUS Authentication
- Single RADIUS Server Configuration
- Multiple RADIUS Server Configuration
- RADIUS Server Group Configuration
- RADIUS Server Configuration using Server Groups

RADIUS Authentication

Example The following example shows how to configure the switch to authenticate using RADIUS:

Figure 43-6: Sample RADIUS Authentication to configure the switch to authenticate users

```
!
radius-server host 172.10.10.1
radius-server key radiuspass
username newuser password newpass
aaa authentication login admin
!
```

The lines in this example RADIUS authentication and accounting configuration are defined as follows:

- The radius-server host command defines the IP address of the RADIUS server host.
- The radius-server key command defines the shared secret text string between the network access server and the RADIUS server host.
- The aaa authentication login command defines a method list named **admin** for login authentication.

Example The following example shows how to configure the switch to authenticate logins using RADIUS:

Figure 43-7: Sample RADIUS Authentication to authenticate logins

```
!
aaa authentication login radius-login group radius
!
```

This sample RADIUS authentication configuration is defined as follows:

- The **aaa authentication login radius-login group radius** command configures the switch to use RADIUS for authentication at the login prompt.

Example The following example shows how to configure the authentication method to verify a username and password at login. In this example, if a username is entered at the username prompt, that username is used for authentication.

Figure 43-8: Sample RADIUS Authentication to verify a username and password

```
!
aaa authentication login default group radius
radius-server host 172.10.10.1 auth-port 1812 acct-port 1813
!
```

The lines in this sample RADIUS authentication configuration are defined as follows:

- The **aaa authentication login default group radius** command specifies that the username and password are verified by RADIUS.
- The **radius-server host 172.10.10.1 auth-port 1812 acct-port 1813** command specifies the IP address of the RADIUS server host, the UDP destination port for authentication requests, and the UDP destination port for accounting requests.

Single RADIUS Server Configuration

Example The following example shows how to configure server-specific timeout, retransmit, and key values for the RADIUS server with IP address 172.2.2.2:

Figure 43-9: Single RADIUS Server sample configuration

```
!
radius-server host 172.2.2.2 timeout 5 retransmit 5 key 10
!
```

Multiple RADIUS Server Configuration

Example The following example shows how to configure two RADIUS servers with specific timeout, retransmit, and key values. The radius-server retransmit command changes the global retransmission value to 4 for all RADIUS servers. The radius-server host command configures specific timeout, retransmission, and key values for the RADIUS server hosts with IP addresses `172.2.2.2` and `172.1.1.1`

Figure 43-10: Multiple RADIUS Server sample configuration

```
!
! Enable and configure radius authentication and accounting
! services on the switch:
!
aaa authentication login default group radius
aaa accounting default start-stop group radius
!
! Change the retransmission value for all RADIUS servers:
!
radius-server retransmit 4
!
! Configure per-server specific timeout, retransmission, and
! key values. Change the default auth-port and acct-port
! values.
!
radius-server host 172.2.2.2 auth-port 1645 acct-port 1646
timeout 3 retransmit 3 key radkey
!
! Configure per-server specific timeout and key values. This
! server uses the global retransmission value.
!
radius-server host 172.1.1.1 timeout 6 key rad123
!
```

RADIUS Server Group Configuration

Example The following example shows how to create server group `group2` with three RADIUS server members, each with the same IP address but with unique authentication and accounting ports:

Figure 43-11: RADIUS Server Group sample configuration using the same IP address

```
!
aaa group server radius group2
  server 172.1.1.1 auth-port 1645 acct-port 1646
  server 172.1.1.1 auth-port 1812 acct-port 1813
  server 172.1.1.1 auth-port 2000 acct-port 2001
!
```

RADIUS Server Configuration using Server Groups

The following example shows how to configure the network access server to recognize two different RADIUS server groups.

One of these groups, `group1`, has two different host entries on the same RADIUS server configured for the same services. The second host entry configured acts as fail over backup to the first one. Each group is individually configured for `deadtime`; `deadtime` for `group1` is one minute, and `deadtime` for `group2` is two minutes.

Figure 43-12: Multiple RADIUS Servers using Server Groups sample configuration

```
!
! The following command configures default RADIUS parameters:
!
aaa authentication login default group group1
!
! The following commands define the group1 RADIUS server group
! and associate servers with it and configures a deadtime of
! one minute:
!
aaa group server radius group1
  server 172.1.1.1 auth-port 1645 acct-port 1646
  server 172.2.2.2 auth-port 1812 acct-port 1813
  deadtime 1
!
! The following commands define the group2 RADIUS server group
! and associate servers with it and configures a deadtime of
! two minutes:
!
aaa group server radius group2
  server 172.2.2.2 auth-port 1812 acct-port 1813
  server 172.3.3.3 auth-port 2000 acct-port 2001
  deadtime 2
!
! The following commands configure the RADIUS attributes
! for each host entry associated with one of the defined
! server groups:
!
radius-server host 172.1.1.1 auth-port 1645 acct-port 1646
radius-server host 172.2.2.2 auth-port 1812 acct-port 1813
radius-server host 172.3.3.3 auth-port 2000 acct-port 2001
!
```

Chapter 44: RADIUS Commands

Command List

This chapter provides an alphabetical reference for commands used to configure the device to use RADIUS servers.

deadtime (RADIUS server group)

Use this command to configure the **deadtime** parameter for the RADIUS server group. This command overrides the global dead-time configured by the radius-server deadtime command on page 44.5. The configured deadtime is the time period in minutes to skip a RADIUS server for authentication or accounting requests if the server is "dead". Note that a RADIUS server is considered "dead" if there is no response from the server within a defined time period.

Use the **no** variant of this command to reset the deadtime configured for the RADIUS server group. If the global deadtime for RADIUS server is configured the value will be used for the servers in the group. The global deadtime for the RADIUS server is set to 0 minutes by default.

Syntax

```
deadtime <0-1440>

no deadtime
```

Parameter	Description
<0-1440>	Amount of time in minutes.

Default The deadtime is set to 0 minutes by default.

Mode Server Group Configuration

Usage If the RADIUS server does not respond to a request packet, the packet is retransmitted the number of times configured for the **retransmit** parameter (after waiting for a **timeout** period to expire). The server is then marked "dead", and the time is recorded. The **deadtime** parameter configures the amount of time to skip a dead server; if a server is dead, no request message is sent to the server for the **deadtime** period.

Examples To configure the deadtime for 5 minutes for the RADIUS server group "GROUP1", use the command:

```
   awplus(config)# aaa group server radius GROUP1
awplus(config-sg)# server 192.168.1.1
awplus(config-sg)# deadtime 5
```

To remove the deadtime configured for the RADIUS server group "GROUP1", use the command:

```
   awplus(config)# aaa group server radius GROUP1
awplus(config-sg)# no deadtime
```

Related Commands aaa group server
radius-server deadtime

debug radius

This command enables RADIUS debugging. If no option is specified, all debugging options are enabled.

Use the **no** variant of this command to disable RADIUS debugging. If no option is specified, all debugging options are disabled.

Syntax

```
debug radius [packet|event|all]

no debug radius [packet|event|all]
```

Parameter	Description
`packet`	Debugging for RADIUS packets is enabled or disabled.
`event`	Debugging for RADIUS events is enabled or disabled.
`all`	Enable or disable all debugging options.

Default RADIUS debugging is disabled by default.

Mode Privileged Exec

Examples To enable debugging for RADIUS packets, use the command:

```
awplus#  debug radius packet
```

To enable debugging for RADIUS events, use the command:

```
awplus#  debug radius event
```

To disable debugging for RADIUS packets, use the command:

```
awplus# no debug radius packet
```

To disable debugging for RADIUS events, use the command:

```
awplus# no debug radius event
```

Related Commands show debugging radius
undebug radius

ip radius source-interface

This command configures the source IP address of every outgoing RADIUS packet to use a specific IP address or the IP address of a specific interface. If the specified interface is down or there is no IP address on the interface, then the source IP address of outgoing RADIUS packets depends on the interface the packets leave.

Use the **no** variant of this command to remove the source interface configuration. The source IP address in outgoing RADIUS packets will be the IP address of the interface from which the packets are sent.

Syntax

```
ip radius source-interface {<interface>|<ip-address>}
```

```
no ip radius source-interface
```

Parameter	Description
`<interface>`	Interface name.
`<ip-address>`	IP address in the dotted decimal format A.B.C.D.

Default Source IP address of outgoing RADIUS packets depends on the interface the packets leave.

Mode Global Configuration

Examples To configure all outgoing RADIUS packets to use the IP address of the interface "vlan1" for the source IP address, use the following commands:

```
        awplus# configure terminal
awplus(config)# ip radius source-interface vlan1
```

To configure the source IP address of all outgoing RADIUS packets to use 192.168.1.10, use the following commands:

```
        awplus# configure terminal
awplus(config)# ip radius source-interface 192.168.1.10
```

To reset the source interface configuration for all outgoing RADIUS packets, use the following commands:

```
        awplus# configure terminal
awplus(config)# no ip radius source-interface
```

Related Commands radius-server host
show radius statistics

radius-server deadtime

Use this command to specify the global **deadtime** for all RADIUS servers. If a RADIUS server is considered dead, it is skipped for the specified deadtime. This command specifies for how many minutes a RADIUS server that is not responding to authentication requests is passed over by requests for RADIUS authentication.

Use the **no** variant of this command to reset the global deadtime to the default of 0 seconds, so that RADIUS servers are not skipped even if they are dead.

Syntax

```
radius-server deadtime <minutes>
```

```
no radius-server deadtime
```

Parameter	Description
`<minutes>`	RADIUS server deadtime in minutes in the range 0 to 1440 (24 hours).

Default The default RADIUS deadtime configured on the system is 0 seconds.

Mode Global Configuration

Usage The RADIUS client considers a RADIUS server to be dead if it fails to respond to a request after it has been retransmitted as often as specified globally by the radius-server retransmit command or for the server by the radius-server host command. To improve RADIUS response times when some servers may be unavailable, set a **deadtime** to skip dead servers.

Examples To set the dead time of the RADIUS server to 60 minutes, use the following commands:

```
        awplus# configure terminal
awplus(config)# radius-server deadtime 60
```

To disable the dead time of the RADIUS server, use the following commands:

```
        awplus# configure terminal
awplus(config)# no radius-server deadtime
```

Related Commands

deadtime (RADIUS server group)
radius-server host
radius-server retransmit
show radius statistics

radius-server host

Use this command to specify a remote RADIUS server host for authentication or accounting, and to set server-specific parameters. The parameters specified with this command override the corresponding global parameters for RADIUS servers. This command specifies the IP address or host name of the remote RADIUS server host and assigns authentication and accounting destination UDP port numbers.

This command adds the RADIUS server address and sets parameters to the RADIUS server. The RADIUS server is added to the running configuration after you issue this command. If parameters are not set using this command then common system settings are applied.

Use the **no** variant of this command to remove the specified server host as a RADIUS authentication and/or accounting server and set the destination port to the default RADIUS server port number (1812).

Syntax

```
radius-server host {<host-name>|<ip-address>} [acct-port <0-65535>]
    [auth-port <0-65535>] [key <key-string>] [retransmit <0-100>]
    [timeout <1-1000>]
```

```
no radius-server host {<host-name>|<ip-address>}
    [acct-port <0-65535>] [auth-port <0-65535>]
```

Parameter	Description
`<host-name>`	Server host name. The DNS name of the RADIUS server host.
`<ip-address>`	The IP address of the RADIUS server host.
`acct-port`	Accounting port. Specifies the UDP destination port for RADIUS accounting requests. If 0 is specified, the server is not used for accounting. The default UDP port for accounting is 1813.
`<0-65535>`	UDP port number (Accounting port number is set to 1813 by default) Specifies the UDP destination port for RADIUS accounting requests. If 0 is specified, the host is not used for accounting.
`auth-port`	Authentication port. Specifies the UDP destination port for RADIUS authentication requests. If 0 is specified, the server is not used for authentication. The default UDP port for authentication is 1812.
`<0-65535>`	UDP port number (Authentication port number is set to 1812 by default) Specifies the UDP destination port for RADIUS authentication requests. If 0 is specified, the host is not used for authentication.
`timeout`	Specifies the amount of time to wait for a response from the server. If this parameter is not specified the global value configured by the **radius-server timeout** command is used.
`<1-1000>`	Time in seconds to wait for a server reply (timeout is set to 5 seconds by default) The time interval (in seconds) to wait for the RADIUS server to reply before retransmitting a request or considering the server dead. This setting overrides the global value set by the **radius-server timeout** command. If no timeout value is specified for this server, the global value is used.

Parameter(cont.)	Description(cont.)
`retransmit`	Specifies the number of retries before skip to the next server. If this parameter is not specified the global value configured by the **radius-server retransmit** command is used.
<0-100>	Maximum number of retries (maximum number of retries is set to 3 by default) The maximum number of times to resend a RADIUS request to the server, if it does not respond within the timeout interval, before considering it dead and skipping to the next RADIUS server. This setting overrides the global setting of the **radius-server retransmit** command. If no retransmit value is specified, the global value is used.
`key`	Set shared secret key with RADIUS servers
`<key-string>`	Shared key string applied Specifies the shared secret authentication or encryption key for all RADIUS communications between this device and the RADIUS server. This key must match the encryption used on the RADIUS daemon. All leading spaces are ignored, but spaces within and at the end of the string are used. If spaces are used in the string, do not enclose the string in quotation marks unless the quotation marks themselves are part of the key. This setting overrides the global setting of the **radius-server key c**ommand. If no key value is specified, the global value is used.

Default The RADIUS client address is not configured (null) by default. No RADIUS server is configured.

Mode Global Configuration

Usage Multiple **radius-server host** commands can be used to specify multiple hosts. The software searches for hosts in the order they are specified. If no host-specific timeout, retransmit, or key values are specified, the global values apply to that host. If there are multiple RADIUS servers for this client, use this command multiple times—once to specify each server.

If you specify a host without specifying the auth port or the acct port, it will by default be configured for both authentication and accounting, using the default UDP ports. To set a host to be a RADIUS server for authentication requests only, set the **acct-port** parameter to 0; to set the host to be a RADIUS server for accounting requests only, set the auth-port parameter to 0.

A RADIUS server is identified by IP address, authentication port and accounting port. A single host can be configured multiple times with different authentication or accounting ports. All the RADIUS servers configured with this command are included in the predefined RADIUS server group radius, which may be used by AAA authentication, authorization and accounting commands. The client transmits (and retransmits, according to the **retransmit** and **timeout** parameters) RADIUS authentication or accounting requests to the servers in the order you specify them, until it gets a response.

Examples To add the RADIUS server `10.0.0.20`, use the following commands:

```
        awplus#  configure terminal
awplus(config)#  radius-server host 10.0.0.20
```

To set the secret key to **allied** on the RADIUS server 10.0.0.20, use the following commands:

```
        awplus#  configure terminal
awplus(config)#  radius-server host 10.0.0.20 key allied
```

To delete the RADIUS server 10.0.0.20, use the following commands:

```
        awplus#  configure terminal
awplus(config)#  no radius-server host 10.0.0.20
```

To configure rad1.company.com for authentication only, use the following commands:

```
        awplus#  configure terminal
awplus(config)#  radius-server host rad1.company.com
                 acct-port 0
```

To remove the RADIUS server rad1.company.com configured for authentication only, use the following commands:

```
        awplus#  configure terminal
awplus(config)#  no radius-server host rad1.company.com
                 acct-port 0
```

To configure rad2.company.com for accounting only, use the following commands:

```
        awplus#  configure terminal
awplus(config)#  radius-server host rad2.company.com
                 auth-port 0
```

To configure 192.168.1.1 with authentication port 1000, accounting port 1001 and retransmit count 5, use the following commands:

```
        awplus#  configure terminal
awplus(config)#  radius-server host 192.168.1.1 auth-port 1000
                 acct-port 1001 retransmit 5
```

Related Commands

aaa group server
radius-server key
radius-server retransmit
radius-server timeout
show radius statistics

radius-server key

This command sets a global secret key for RADIUS authentication on the switch. The shared secret text string is used for RADIUS authentication between the switch and a RADIUS server.

Note that if no secret key is explicitly specified for a RADIUS server, the global secret key will be used for the shared secret for the server.

Use the **no** variant of this command to reset the secret key to the default (null).

Syntax

```
radius-server key <key>

no radius-server key
```

Parameter	Description
<key>	Shared secret among radius server and 802.1X client.

Default The RADIUS server secret key on the system is not set by default (null).

Mode Global Configuration

Usage Use this command to set the global secret key shared between this client and its RADIUS servers. If no secret key is specified for a particular RADIUS server using the **radius-server host c**ommand, this global key is used.

After enabling AAA authentication with the **aaa authentication login** command, set the authentication and encryption key using the **radius-server key** command so the key entered matches the key used on the RADIUS server.

Examples To set the global secret key to **allied** for RADIUS server, use the following commands:

```
        awplus# configure terminal
awplus(config)# radius-server key allied
```

To set the global secret key to **secret** for RADIUS server, use the following commands:

```
        awplus# configure terminal
awplus(config)# radius-server key secret
```

To delete the global secret key for RADIUS server, use the following commands:

```
        awplus# configure terminal
awplus(config)# no radius-server key
```

Related Commands radius-server host
show radius statistics

radius-server retransmit

This command sets the retransmit counter to use RADIUS authentication on the switch. This command specifies how many times the switch transmits each RADIUS request to the RADIUS server before giving up.

This command configures the **retransmit** parameter for RADIUS servers globally. If the **retransmit** parameter is not specified for a RADIUS server by the **radius-server host** command then the global configuration set by this command is used for the server instead.

Use the **no** variant of this command to reset the re-transmit counter to the default (3).

Syntax

```
radius-server retransmit <retries>
```

```
no radius-server retransmit
```

Parameter	Description
`<retries>`	RADIUS server retries in the range <0-100> The number of times a request is resent to a RADIUS server that does not respond, before the server is considered dead and the next server is tried. If no retransmit value is specified for a particular RADIUS server using the **radius-server host** command, this global value is used.

Default The default RADIUS retransmit count on the switch is 3.

Mode Global Configuration

Examples To set the RADIUS **retransmit** count to 1, use the following commands:

```
        awplus# configure terminal
awplus(config)# radius-server retransmit 1
```

To set the RADIUS **retransmit** count to the default (3), use the following commands:

```
        awplus# configure terminal
awplus(config)# no radius-server retransmit
```

To configure the RADIUS **retransmit** count globally with 5, use the following commands:

```
        awplus# configure terminal
awplus(config)# radius-server retransmit 5
```

To disable retransmission of requests to a RADIUS server, use the following commands:

```
        awplus# configure terminal
awplus(config)# radius-server retransmit 0
```

Related Commands radius-server deadtime
radius-server host
show radius statistics

radius-server timeout

Use this command to specify the RADIUS global timeout value. This is how long the device waits for a reply to a RADIUS request before retransmitting the request, or considering the server to be dead. If no timeout is specified for the particular RADIUS server by the **radius-server host** command, it uses this global timeout value.

Note that this command configures the **timeout** parameter for RADIUS servers globally.

The **no** variant of this command resets the transmit timeout to the default (5 seconds).

Syntax

```
radius-server timeout <seconds>

no radius-server timeout
```

Parameter	Description
`<seconds>`	RADIUS server timeout in seconds in the range 1 to 1000. The global time in seconds to wait for a RADIUS server to reply to a request before retransmitting the request, or considering the server to be dead (depending on the **radius-server retransmit** command).

Default The default RADIUS transmit timeout on the system is 5 seconds.

Mode Global Configuration

Examples To globally set the device to wait 20 seconds before retransmitting a RADIUS request to unresponsive RADIUS servers, use the following commands:

```
        awplus# configure terminal
awplus(config)# radius-server timeout 20
```

To set the RADIUS **timeout** parameter to 1 second, use the following commands:

```
        awplus# configure terminal
awplus(config)# radius-server timeout 1
```

To set the RADIUS **timeout** parameter to the default (5 seconds), use the following commands:

```
        awplus# configure terminal
awplus(config)# no radius-server timeout
```

To configure the RADIUS server **timeout** period globally with 3 seconds, use the following commands:

```
        awplus# configure terminal
awplus(config)# radius-server timeout 3
```

To reset the global **timeout** period for RADIUS servers to the default, use the following command:

```
        awplus# configure terminal
awplus(config)# no radius-server timeout
```

Related Commands

radius-server deadtime
radius-server host
radius-server retransmit
show radius statistics

server (Server Group)

This command adds a RADIUS server to a server group in Server-Group Configuration mode. The RADIUS server should be configured by the radius-server host command.

The server is appended to the server list of the group and the order of configuration determines the precedence of servers. If the server exists in the server group already, it will be removed before added as a new server.

The server is identified by IP address and authentication and accounting UDP port numbers. So a RADIUS server can have multiple entries in a group with different authentication and/or accounting UDP ports. The **auth-port** specifies the UDP destination port for authentication requests to the server. To disable authentication for the server, set `auth-port` to 0. If the authentication port is missing, the default port number is 1812. The **acct-port** specifies the UDP destination port for accounting requests to the server. To disable accounting for the server, set `acct-port` to 0. If the accounting port is missing, the default port number is 1812.

Use the **no** variant of this command to remove a RADIUS server from the server group.

Syntax

```
server {<hostname>|<ip-address>}
    [auth-port <0-65535>][acct-port <0-65535>]

no server {<hostname>|<ip-address>}
    [auth-port <0-65535>][acct-port <0-65535>]
```

Parameter	Description
`<hostname>`	Server host name
`<ip-address>`	Server IP address The server is identified by IP address, authentication and accounting UDP port numbers. So a RADIUS server can have multiple entries in a group with different authentication and/or accounting UDP ports.
`auth-port`	Authentication port The **auth-port** specifies the UDP destination port for authentication requests to the server. To disable authentication for the server, set **auth-port** to 0. If the authentication port is missing, the default port number is 1812.
`<0-65535>`	UDP port number (default: 1812)
`acct-port`	Accounting port The **acct-port** specifies the UDP destination port for accounting requests to the server. To disable accounting for the server, set **acct-port** to 0. If the accounting port is missing, the default port number is 1813.
`<0-65535>`	UDP port number (default: 1813)

Default The default Authentication port number is 1812 and the default Accounting port number is 1813.

Mode Server Group Configuration

Usage The RADIUS server to be added must be configured by the **radius-server host** command. In order to add or remove a server, the **auth-port** and **acct-port** parameters in this command must be the same as the corresponding parameters in the **radius-server host** command.

Examples To create a RADIUS server group RAD_AUTH1 for authentication, use the following commands:

```
           awplus# configure terminal
   awplus(config)# aaa group server radius RAD_AUTH1
awplus(config-sg)# server 192.168.1.1 acct-port 0
awplus(config-sg)# server 192.168.2.1 auth-port 1000
                   acct-port 0
```

To create a RADIUS server group RAD_ACCT1 for accounting, use the following commands:

```
           awplus# configure terminal
   awplus(config)# aaa group server radius RAD_ACCT1
awplus(config-sg)# server 192.168.2.1 auth-port 0
                   acct-port 1001
awplus(config-sg)# server 192.168.3.1 auth-port 0
```

To remove server 192.168.3.1 from the existing server group **GROUP1**, use the following commands:

```
           awplus# configure terminal
   awplus(config)# aaa group server radius GROUP1
awplus(config-sg)# no server 192.168.3.1
```

Related Commands

aaa accounting auth-mac default
aaa accounting auth-web default
aaa accounting dot1x
aaa accounting login
aaa authentication auth-mac
aaa authentication auth-web
aaa authentication login
aaa group server
radius-server host

show debugging radius

This command displays the current debugging status for the RADIUS servers.

Syntax

```
show debugging radius
```

Mode User Exec and Privileged Exec

Example To display the current debugging status of RADIUS servers, use the command:

```
awplus# show debugging radius
```

Output Figure 44-1: Example output from the **show debugging radius** command

```
RADIUS debugging status:
RADIUS event debugging is off
RADIUS packet debugging is off
```

show radius

This command displays the current RADIUS server configuration and status.

Syntax `show radius`

Mode User Exec and Privileged Exec

Example To display the current status of RADIUS servers, use the command:

```
awplus# show radius
```

Output Figure 44-2: Example output from the **show radius** command showing RADIUS servers

```
RADIUS Global Configuration
Source Interface : not configured
Secret Key : secret
Timeout : 5 sec
Retransmit Count : 3
Deadtime : 20 min
Server Host : 192.168.1.10
Authentication Port : 1812
Accounting Port : 1813
Secret Key : secret
Timeout : 3 sec
Retransmit Count : 2
Server Host : 192.168.1.11
Authentication Port : 1812
Accounting Port : not configured
Server Name/Auth Acct Auth Acct
IP Address Port Port Status Status
------------------------------------------------------------
192.168.1.10 1812 1813 Alive Alive
192.168.1.11 1812 N/A Alive N/A
```

Example See the sample output below showing RADIUS client status and RADIUS configuration:

```
awplus# show radius
```

Output Figure 44-3: Example output from the **show radius** command showing RADIUS client status

```
RADIUS global interface name: awplus
Secret key:
Timeout: 5
Retransmit count: 3
Deadtime: 0

Server Address: 150.87.18.89
Auth destination port: 1812
Accounting port: 1813
Secret key: swg
Timeout: 5
Retransmit count: 3
Deadtime: 0show radius local-server group
```

Output Parameter	Meaning	
`Source Interface`	The interface name or IP address to be used for the source address of all outgoing RADIUS packets.	
`Secret Key`	A shared secret key to a radius server.	
`Timeout`	A time interval in seconds.	
`Retransmit Count`	The number of retry count if a RADIUS server does not response.	
`Deadtime`	A time interval in minutes to mark a RADIUS server as “dead”.	
`Interim-Update`	A time interval in minutes to send Interim-Update Accounting report.	
`Group Deadtime`	The deadtime configured for RADIUS servers within a server group.	
`Server Host`	The RADIUS server hostname or IP address.	
`Authentication Port`	The destination UDP port for RADIUS authentication requests.	
`Accounting Port`	The destination UDP port for RADIUS accounting requests.	
`Auth Status`	The status of the authentication port. The status ("dead", “error”, or "alive") of the RADIUS authentication server and, if dead, how long it has been dead for.	
	`Alive`	The server is alive.
	`Error`	The server is not responding.
	`Dead`	The server is detected as dead and it will not be used for deadtime period. The time displayed in the output shows the server is in dead status for that amount of time.
	`Unknown`	The server is never used or the status is unknown.
`Acct Status`	The status of the accounting port. The status ("dead", “error”, or "alive") of the RADIUS accounting server and, if dead, how long it has been dead for.	

show radius statistics

This command shows the RADIUS client statistics for the switch.

Syntax

```
show radius statistics
```

Mode User Exec and Privileged Exec

Example See the sample output below showing RADIUS client statistics and RADIUS configuration:

```
awplus# show radius statistics
```

Output Figure 44-4: Example output from the **show radius** statistics command:

```
RADIUS statistics for Server: 150.87.18.89
Access-Request Tx : 5 - Retransmit : 0
Access-Accept Rx : 1 - Access-Reject Rx : 2
Access-Challenge Rx : 2
Unknown Type : 0 - Bad Authenticator: 0
Malformed Access-Resp: 0 - Wrong Identifier: 0
Bad Attribute : 0 - Packet Dropped : 0
TimeOut : 0 - Dead count : 0
Pending Request: 0
```

undebug radius

This command applies the functionality of the no debug radius command on page 44.3.

Chapter 45: TACACS+ Introduction and Configuration

Introduction

This chapter provides information about the AlliedWare Plus implementation of TACACS+ and how to configure it on this switch. For detailed descriptions of the commands used to configure TACACS+, see Chapter 46, TACACS+ Commands. For information about Authentication, Authorization and Accounting (AAA), see Chapter 41, AAA Introduction and Configuration and Chapter 42, AAA Commands.

TACACS+ Overview

TACACS+ (Terminal Access Controller Access-Control System Plus) provides a method for securely managing multiple network access points from a single management service.

TACACS+ is a TCP-based access control protocol, utilizing TCP port 49, that allows a device to forward a user's username and password to an authentication server to determine whether access can be allowed. In addition to this authentication service, TACACS+ can also provide authorization and accounting services.

One of the features of TACACS+ is the ability to separate authentication, authorization and accounting so that these functions can be provided independently on separate servers. Authentication involves identifying a user, typically by requiring the user to supply a valid username and password before access is granted. Following authentication, the user must gain authorization to perform tasks. For example, after logging into a switch, a user may try to issue configuration commands. The authorization process determines whether the user has the authority to issue these commands. Authorization is always preceded by authentication.

The AlliedWare Plus TACACS+ Implementation

The AlliedWare Plus TACACS+ implementation provides authentication, authorization, and accounting. Note that:

- Authorization cannot be performed independently of the authentication process. There are no authorization commands available.
- Authentication and authorization must be configured on the same server.
- Authorization is only applicable if enable password authentication has not been configured with the **aaa authentication enable default group tacacs+** command.

With the AlliedWare Plus TACACS+ implementation, all traffic that passes between the TACACS+ client and the TACACS+ servers on the network is encrypted. TACACS+ encrypts the entire payload of packets, which means that it encrypts the user's password between the client and the server.

A TACACS+ client is available on your switch. You need a system running TACACS+ server software from a software provider to use the TACACS+ functionality on your switch.

Authentication

The TACACS+ protocol can forward many types of username and password information. The AlliedWare Plus TACACS+ implementation supports username and password login authentication, as well as enable password authentication. This information is encrypted over the network with MD5 (Message Digest 5).

When TACACS+ login authentication is enabled on the switch with the aaa authentication login command and at least one TACACS+ server is configured and reachable, all user login authentications are authenticated against the TACACS+ server. No local login or other means of authentication is allowed or accepted by the switch unless the switch has been configured to use another authentication method as a backup, and the TACACS+ server is not reachable.

When TACACS+ enable password authentication is enabled on the switch with the aaa authentication enable default group tacacs+ command and at least one TACACS+ server is configured and reachable, all user attempts to access a higher privilege level using the enable (Privileged Exec mode) command are authenticated against the TACACS+ server. If TACACS+ enable password authentication is enabled and the TACACS+ server is not reachable, then the user is only granted access to the desired privilege level if a backup authentication method is also configured.

Note If TACACS+ login authentication is enabled on the switch, and enable password authentication is configured as default with the aaa authentication enable default local command, then a local enable password must be configured for each privilege level that needs to be accessible to users.

Authorization

In the AlliedWare Plus TACACS+ implementation, authorization cannot be performed independently of the authentication process. Authorization is concerned with what users are allowed to do once they have gained access to the managed device. This involves the passing of Attribute Value pairs (AV pairs) from the TACACS+ server to the managed device. An AV pair is made up of two pieces of information: the attribute that identifies the parameter to be set, and the value that specifies the value to assign to that parameter. These AV pairs are configured on a per-user or per-group basis on the TACACS+ server. The AV pairs that are supported by the AlliedWare Plus TACACS+ implementation are:

- Privilege Level

  Privilege levels range from 1 to 15, with 15 being the highest. For information about privilege levels see "How to Add and Remove Users" on page 1.32 and the username command on page 5.35.

- Timeout

  The value assigned to this attribute specifies the length of time that the session can exist. After this value has expired, the session will either be disconnected, or have the privilege of the user reduced. The valid range of timeout values is 0 to 65535 (minutes).

- Idletime

  If no input or output traffic is received or sent in the period specified by the value for this attribute, the session is disconnected. The valid idletime range is 0 to 65535 (minutes).

Note In the AlliedWare Plus TACACS+ implementation, authorization for privilege level, timeout, and idletime AV pairs is only attempted if enable password authentication (**aaa authentication enable default group tacacs+** command) is not configured. If enable password authentication is configured then the privilege level a user is granted access to is determined during the enable password authentication session.

Accounting

TACACS+ accounting usually takes place after authentication and authorization. However, because TACACS+ separates these three functions, neither authentication nor authorization are required for accounting to function. TACACS+ accounting provides the following two distinct functions:

- a record of services used for billing purposes
- an audit trail for user exec sessions

The AlliedWare Plus TACACS+ accounting implementation supports an audit trail for user exec sessions only. This includes the ability to configure accounting for user logins and logouts, and accounting of any commands executed by the user while they are logged into the switch.

TACACS+ accounting includes three different types of accounting records:

- **start** records that indicate a service is about to start
- **stop** records that indicate a service has just ended
- **update** records that indicate a service is still in progress

Configuration

This section describes how to set up TACACS+ for login authentication, enable password authentication, and accounting.

The TACACS+ server is normally a multiuser system running TACACS+ server software from a software provider. TACACS+ servers are identified on the basis of their host name or IP address. A TACACS+ server and a switch use a shared secret text string to encrypt passwords and exchange responses. To configure TACACS+, you must specify the host running the TACACS+ server software and a secret text string that it shares with the switch.

Configure TACACS+

Table 45-1: General configuration procedure for TACACS+ authentication and accounting

Specify a remote TACACS+ server and the shared secret key	
awplus# `configure terminal`	Enter Global Configuration mode.
awplus(config)# `tacacs-server host {<host-name>|<ip-address>} [key [8]<key-string>]`	Specify the IP address or host name of the remote TACACS+ server host and the shared secret key to use with the specified TACACS+ server. Specify 8 if you are entering a password as a string that has already been encrypted instead of entering a plain text password. As many as four TACACS+ servers can be configured and consulted for authentication and accounting. The first server configured is regarded as the primary server and if the primary server fails then the backup servers are consulted in turn.
awplus(config)# `tacacs-server key [8] <key-string>`	Specify the global shared secret text string used between the switch and **all** TACACS+ servers. Specify 8 if you are entering a password as a string that has already been encrypted instead of entering a plain text password. If no secret key is explicitly specified for a TACACS+ server with the tacacs-server host command, the global secret key will be used.
Specify the timeout value	
awplus(config)# `tacacs-server timeout <seconds>`	Specify for how many seconds a switch waits for a reply to a TACACS+ request before considering the TACACS+ server dead.
Define the method list for TACACS+ login authentication	
awplus(config)# `aaa authentication login {default|<list-name>} {[local] [group {radius|tacacs+|<group-name>}]}`	This method list defines the AAA server type used for login authentication. The server types are always used in the order specified with this command. If the first server in the method list is unreachable, the switch sends the request to the next server in the list. If the authentication server denies the authentication request because of an incorrect username or password then the user login fails.

Table 45-1: General configuration procedure for TACACS+ authentication and accounting(cont.)

Define the method list for TACACS+ enable password authentication	
awplus(config)# aaa authentication enable default group tacacs+ [local] [none]	This method list defines the authentication method used to determine the privilege command level a user can access. Specify **local** to use the locally configured enable password and **none** to grant access to Privileged Exec mode with no authentication, if the TACACS+ server goes offline, or is not reachable during enable password authentication.
Define the method for TACACS+ login accounting	
awplus(config)# aaa accounting login {default|*<list-name>*} {start-stop|stop-only|none} {group {radius|tacacs+|*<group-name>*}}	You can only define one method for login accounting, either RADIUS or TACACS+. Specify **start-stop** to send both start and stop login accounting records, **stop-only** to send only stop login accounting records, or **none** to disable the sending of login accounting records.
Configure TACACS+ command accounting	
awplus(config)# aaa accounting commands *<1-15>* default stop-only group tacacs+	TACACS+ command accounting is configured per privilege level and only commands of the specified privilege level are accounted. Therefore, if you require that all commands are accounted to the TACACS+ server, you must configure command accounting for each privilege level separately. Commands are accounted to the TACACS+ server after they have successfully executed.
Troubleshooting TACACS+	
awplus(config)# show tacacs+	Display the current TACACS+ server configuration and status.
awplus# debug aaa authentication	Enable debug output for TACACS+ authentication.
awplus# debug aaa authorization	Enable debug output for TACACS+ authorization.
awplus# debug aaa accounting	Enable debug output for TACACS+ accounting.

TACACS+ Configuration Example

Example The following example shows how to configure the switch to authenticate and account using TACACS+.

Figure 45-1: Sample TACACS+ authentication and accounting to configure the switch to authenticate and account user exec sessions

```
!
tacacs-server host 172.10.10.1
tacacs-server key tacacspass
aaa authentication login admin group tacacs+ local
aaa authentication enable default group tacacs+ local
aaa accounting login admin start-stop group tacacs+
aaa accounting commands 1 default stop-only group tacacs+
aaa accounting commands 7 default stop-only group tacacs+
aaa accounting commands 15 default stop-only group tacacs+

line console 0
login authentication admin
accounting login admin
!
```

The lines in this example TACACS+ authentication and accounting configuration are defined as follows:

- The tacacs-server host command defines the IP address of the TACACS+ server host.
- The tacacs-server key command defines the global shared secret text string between the network access server and the TACACS+ server host.
- The aaa authentication login command defines a method list named **admin** to use first the TACACS+ servers and then the local user database for user login authentication.
- The aaa authentication enable default group tacacs+ command defines a method list to use first the TACACS+ servers and then the local enable passwords, set with the enable password command, for user enable password authentication.
- The aaa accounting login command defines a method named **admin** to use TACACS+ servers for login accounting.
- The aaa accounting commands command specifies the privilege level of the commands that will be accounted.
- The login authentication command specifies that this method list will be used for authenticating users logging in on the asynchronous console port.
- The accounting login command specifies that this method list will be used for accounting users logging in on the asynchronous console port.

Chapter 46: TACACS+ Commands

Command List

This chapter provides an alphabetical reference for commands used to configure the device to use TACACS+ servers. For more information about TACAC+, see Chapter 45, TACACS+ Introduction and Configuration.

tacacs-server host

Use this command to specify a remote TACACS+ server host for authentication, authorization and accounting, and to set the shared secret key to use with the TACACS+ server. The parameters specified with this command override the corresponding global parameters for TACACS+ servers.

Use the **no** variant of this command to remove the specified server host as a TACACS+ authentication and authorization server.

Syntax

```
tacacs-server host {<host-name>|<ip-address>} [key [8]<key-string>]

no tacacs-server host {<host-name>|<ip-address>}
```

Parameter	Description
`<host-name>`	Server host name. The DNS name of the TACACS+ server host.
`<ip-address>`	The IP address of the TACACS+ server host, in dotted decimal notation A.B.C.D.
`key`	Set shared secret key with TACACS+ servers.
8	Specifies that you are entering a password as a string that has already been encrypted instead of entering a plain text password. The running config displays the new password as an encrypted string even if password encryption is turned off.
`<key-string>`	Shared key string applied, a value in the range 1 to 64 characters. Specifies the shared secret authentication or encryption key for all TACACS+ communications between this device and the TACACS+ server. This key must match the encryption used on the TACACS+ server. This setting overrides the global setting of the tacacs-server key command. If no key value is specified, the global value is used.

Default No TACACS+ server is configured by default.

Mode Global Configuration

Usage A TACACS+ server host cannot be configured multiple times like a RADIUS server.

As many as four TACACS+ servers can be configured and consulted for login authentication, enable password authentication and accounting. The first server configured is regarded as the primary server and if the primary server fails then the backup servers are consulted in turn. A backup server is consulted if the primary server fails, not if a login authentication attempt is rejected. The reasons a server would fail are:

- it is not network reachable
- it is not currently TACACS+ capable
- it cannot communicate with the switch properly due to the switch and the server having different secret keys

Examples To add the server `tac1.company.com` as the TACACS+ server host, use the following commands:

```
        awplus# configure terminal
awplus(config)# tacacs-server host tac1.company.com
```

To set the secret key to `secret` on the TACACS+ server `192.168.1.1`, use the following commands:

```
        awplus# configure terminal
awplus(config)# tacacs-server host 192.168.1.1 key secret
```

To remove the TACACS+ server `tac1.company.com`, use the following commands:

```
        awplus# configure terminal
awplus(config)# no tacacs-server host tac1.company.com
```

Related Commands aaa accounting commands
aaa authentication login
tacacs-server key
tacacs-server timeout
show tacacs+

tacacs-server key

This command sets a global secret key for TACACS+ authentication, authorization and accounting. The shared secret text string is used for TACACS+ communications between the switch and all TACACS+ servers.

Note that if no secret key is explicitly specified for a TACACS+ server with the tacacs-server host command, the global secret key will be used for the shared secret for the server.

Use the **no** variant of this command to remove the global secret key.

Syntax

```
tacacs-server key [8] <key-string>

no tacacs-server key
```

Parameter	Description
8	Specifies a string in an encrypted format instead of plain text. The running config will display the new password as an encrypted string even if password encryption is turned off.
<key-string>	Shared key string applied, a value in the range 1 to 64 characters. Specifies the shared secret authentication or encryption key for all TACACS+ communications between this device and all TACACS+ servers. This key must match the encryption used on the TACACS+ server.

Mode Global Configuration

Usage Use this command to set the global secret key shared between this client and its TACACS+ servers. If no secret key is specified for a particular TACACS+ server using the tacacs-server host command, this global key is used.

Examples To set the global secret key to `secret` for TACACS+ server, use the following commands:

```
        awplus# configure terminal
awplus(config)# tacacs-server key secret
```

To delete the global secret key for TACACS+ server, use the following commands:

```
        awplus# configure terminal
awplus(config)# no tacacs-server key
```

Related Commands tacacs-server host
show tacacs+

tacacs-server timeout

Use this command to specify the TACACS+ global timeout value. The timeout value is how long the device waits for a reply to a TACACS+ request before considering the server to be dead.

Note that this command configures the **timeout** parameter for TACACS+ servers globally.

The **no** variant of this command resets the transmit timeout to the default (5 seconds).

Syntax

```
tacacs-server timeout <seconds>

no tacacs-server timeout
```

Parameter	Description
<seconds>	TACACS+ server timeout in seconds, in the range 1 to 1000.

Default The default timeout value is 5 seconds.

Mode Global Configuration

Examples To set the timeout value to 3 seconds, use the following commands:

```
        awplus# configure terminal
awplus(config)# tacacs-server timeout 3
```

To reset the timeout period for TACACS+ servers to the default, use the following commands:

```
        awplus# configure terminal
awplus(config)# no tacacs-server timeout
```

Related Commands tacacs-server host
show tacacs+

show tacacs+

This command displays the current TACACS+ server configuration and status.

Syntax

```
show tacacs+
```

Mode User Exec and Privileged Exec

Example To display the current status of TACACS+ servers, use the command:

```
awplus# show tacacs+
```

Output Figure 46-1: Example output from the **show tacacs+** command

```
TACACS+ Global Configuration
  Timeout                   : 5 sec

Server Host/       Server
IP Address         Status
-------------------------------------------------------------
192.168.1.10       Alive
192.168.1.11       Unknown
```

Table 46-1: Parameters in the output of the **show tacacs+** command

Output Parameter	Meaning	
`Timeout`	A time interval in seconds.	
`Server Host/IP Address`	TACACS+ server hostname or IP address.	
`Server Status`	The status of the authentication port.	
	`Alive`	The server is alive.
	`Dead`	The server has timed out.
	`Error`	The server is not responding or there is an error in the key string entered.
	`Unknown`	The server is never used or the status is unkown.
	`Unreachable`	The server is unreachable.
	`Unresolved`	The server name can not be resolved.

Chapter 47: Local RADIUS Server Introduction and Configuration

Local RADIUS Server Introduction

Local RADIUS Server provides a user authentication service feature. This feature must be enabled on the switch, because it is disabled by default. For details of commands used to configure the local RADIUS server, see Chapter 48, Local RADIUS Server Commands.

Enable the Local RADIUS Server

The Local RADIUS Server is disabled by default. Enter the following commands to enable the Local RADIUS Server:

```
              awplus# configure terminal
      awplus(config)# radius-server local
awplus(config-radsrv)# server enable
```

This will automatically initialize the internal Certificate Authority (CA) in the switch. It will also automatically create a server certificate and enrol the certificate with the Local CA by implicitly executing the following commands:

```
      awplus(config)# crypto pki trustpoint local
      awplus(config)# crypto pki enroll local
```

The **crypto pki trustpoint local** command declares the Local CA as the CA from which to obtain Certificates. The Local CA has be defined first so Certificates can be obtained from it. The crypto pki enroll local command obtains the system certificate from the Local CA.

The switch is automatically added to the list of authenticators that may send authentication requests to the Local RADIUS Server by implicitly executing the following commands:

```
              awplus# configure terminal
      awplus(config)# radius-server local
awplus(config-radsrv)# nas 127.0.0.1 key awplus-local-radius-
                      server
```

Note The key **awplus-local-radius-server** is a pre-defined component that can be used for internal exchanges between the switch's RADIUS client and its RADIUS server.

Add the Local RADIUS Server as a RADIUS Server

Although the switch is automatically defined as a NAS (Network Access Server) for the Local RADIUS Server, you must manually add the Local RADIUS Server to the server list defined for the Local RADIUS Client.

Use the following commands to add the Local RADIUS Server as a RADIUS Server. The Local RADIUS Client can then send authentication requests to its Local RADIUS Server:

```
        awplus# configure terminal

awplus(config)# radius-server host 127.0.0.1 key awplus-local-
                radius-server
```

Add authenticators to the list of authenticators

Authenticators can send authentication requests to the Local RADIUS Server.

Use the following commands to add other authenticators to the list of authenticators.

```
               awplus# configure terminal

       awplus(config)# radius-server local

awplus(config-radsrv)# nas <nas-ip-address> key <nas-keystring>
```

Configure the Local RADIUS Server User Database

Add users to the RADIUS user list without assigning VLANs

For entries that will be used to authenticate dot1x supplicants, but not assign them to a VLAN, the following commands will add users to the RADIUS user list:

```
            awplus#  configure terminal

    awplus(config)#  radius-server local

awplus(config-radsrv)#  user <radius-user-name> password <user-
                        password>
```

Add users to the RADIUS user list and assign VLANs

Add users to the RADIUS user list, and define a VLAN ID that will be assigned to them.

To add entries to be used to authenticate dot1x supplicants, and assign them to a VLAN, follow the two steps shown below:

Step 1: Create groups associated with the VIDs that will be allocated

Enter the following commands to create groups with the VIDs that will be allocated to them:

```
            awplus#  configure terminal

    awplus(config)#  radius-server local

awplus(config-radsrv)#  group VLAN10Users

awplus(config-radsrv)#  vlan 10

awplus(config-radsrv)#  group VLAN11Users

awplus(config-radsrv)#  vlan 11
```

Step 2: Add the users after creating groups

Add the users and refer to the relevant group in the command that creates the user as below:

```
            awplus#  configure terminal

    awplus(config)#  radius-server local

awplus(config-radsrv)#  user VCSPCVLAN10 password VCSPCPass
                        group VLAN10Users

awplus(config-radsrv)#  user VCSPCVLAN11 password VCSPCPass
                        group VLAN11Users
```

Authenticating login sessions

Authentication can be performed in multiple contexts, such as the authentication of users logging in at a console, as well as tri-authentication of devices connecting to switch ports, see Tri-Authentication Configuration in Chapter 39, Authentication Introduction and Configuration.

RADIUS Authentication with User Privileges

A new security privilege level is available from release 5.4.1. Users with privilege levels 1 to 6 will continue to have access to privilege 1 level commands. Users with privilege 7 to 14 will have access to level 1 commands plus all show commands. Users with privilege level 15 will continue to have access to all commands.

As a result of this new intermediate access level, when a user logs into a management session on a switch running release 5.4.1, and is being authenticated by RADIUS, the RADIUS server needs to be able to indicate to the switch what privilege level to assign to the user's session.

In particular, this means that extra configuration is now required when the RADIUS Server is being set up with users for who will be logging into the switch by console, telnet, or SSH.

The way that the privilege level is associated with a user is, of course, to use the RADIUS attributes. The attributes are configured on RADIUS groups.

Since there are now three security privilege levels there will need to be up to three different groups for login users; each group specifying a different privilege level.

The attributes that need to be configured on the three different RADIUS groups are as follows:

1. For the users with a privilege level of 1-6 use just the RADIUS `attribute Service-Type`, and assign it the value `NAS-Prompt-User`:

```
attribute Service-Type NAS-Prompt-User
```

2. For users with the security privilege of 7-14 use the following 2 RADIUS attributes:

```
attribute Cisco-AVPair shell:priv-lvl=7
attribute Service-Type NAS-Prompt-User
```

3. User with the administrator security privilege use just the RADIUS `attribute Service-Type`, and assign it the value `Administrative-User`:

```
attribute Service-Type Administrative-User
```

Since there is not an explicit RADIUS attribute for the users with the security privilege level 7, we use "`Cisco-AVPair`" to specify this user privilege. But it's very important that we also specify the attribute `Service-Type NAS-Prompt-User` as well, otherwise the following error is generated when a user allocated to this group tries to login into the AlliedWare Plus switch:

```
19:09:14 awplus login[16974]: Invalid user name "tests" in main:698.
Abort.
```

The RADIUS Server attribute `NAS-Prompt-User` is used for non-privileged level users as per the RADIUS RFC. This attribute is used for users with security privilege levels of 1 to 6.

Configuring these RADIUS Server attributes is achieved using Local RADIUS Server commands:

```
                awplus# configure terminal
        awplus(config)# radius-server local
 awplus(config-radsrv)# group users
awplus(config-radsrv-group)# attribute Service-Type NAS-
                             Prompt_User
```

See the below sample configuration for an AlliedWare Plus switch acting as the RADIUS Server, with the three different security privileges for `admin`, `middle-management`, and `users` groups:

Figure 47-1: Sample RADIUS Server configuration for three different security privileges:

```
crypto pki trustpoint local
!
crypto pki enroll local
radius-server local
 server enable
 nas 10.1.1.1 key test
 nas 127.0.0.1 key awplus-local-radius-server
 group admin
  attribute Service-Type Administrative-User
 group middle-management
  attribute Cisco-AVPair shell:priv-lvl=7
  attribute Service-Type NAS-Prompt-User
 group users
  attribute Service-Type NAS-Prompt-User
 user test encrypted password UukoSyvxY2v9iWXm8e/
JMDJd9iIc3RPyY091GOb3pA4= group  users
 user tested encrypted password
sEDhM4iJRfJrLhhs+RgjpgkDXtCwuji6AllpApi9EjA= group admin
 user tests encrypted password il9aIh8JLOT6kHDV+Ix7/
8fzyfVpAwRErJg6NPQdJy8= group middle-management
```

Removing users from the RADIUS users list

To remove the user Tom from the user database of the Local RADIUS server, use the commands:

```
               awplus# configure terminal
       awplus(config)# radius-server local
awplus(config-radsrv)# no user Tom
```

Creating certificates for single users and all users

Create a certificate for a single user

A certificate for user Tom can be created from the local CA by using the commands:

```
        awplus# configure terminal
awplus(config)# crypto pki enroll local user Tom
```

Create a certificate for all users

Certificates can be created for all currently defined users by using the commands:

```
        awplus# configure terminal
awplus(config)# crypto pki enroll local local-radius-all-users
```

Exporting certificates

User certificates can be exported in PKCS12 format.

To export a certificate for user Tom and upload it to the TFTP server at 192.168.1.1, use the commands:

```
        awplus# configure terminal
awplus(config)# crypto pki export local pkcs12 Tom tftp://
                192.168.1.1/tomcert.pkcs
```

Defined RADIUS attributes list

This is a full list of valid attributes and pre-defined values that may be used in conjunction with the **attribute** command on page 48.2, to show or configure defined RADIUS attributes.

Table 47-1 lists all Standard attributes and values, Table 47-2 lists the Vendor-Specific attribute (attribute ID 26) names and values.

More detailed information can be found in the following RFCs, defining the attributes and values for RADIUS server:

- RFC2865: Remote Authentication Dial In User Service (RADIUS)
- RFC2866: RADIUS Accounting
- RFC2867: RADIUS Accounting Modifications for Tunnel Protocol Support
- RFC2868: RADIUS Attributes for Tunnel Protocol Support
- RFC2869: RADIUS Extensions
- RFC3162: RADIUS and IPv6
- RFC3576: Dynamic Authorization Extensions to Remote Authentication Dial In User Service (RADIUS)
- RFC3580: IEEE 802.1X Remote Authentication Dial In User Service (RADIUS) Usage Guidelines
- RFC4072: Diameter Extensible Authentication Protocol (EAP) Application
- RFC4372: Chargeable User Identity
- RFC4603: Additional Values for the NAS-Port-Type Attribute
- RFC4675: RADIUS Attributes for Virtual LAN and Priority Support
- RFC4679: DSL Forum Vendor-Specific RADIUS Attributes
- RFC4818: RADIUS Delegated-IPv6-Prefix Attribute
- RFC4849: RADIUS Filter Rule Attribute
- RFC5176: Dynamic Authorization Extensions to Remote Authentication Dial In User Service (RADIUS)
- RFC5580: Carrying Location Objects in RADIUS and Diameter
- RFC5607: Remote Authentication Dial-In User Service (RADIUS) Authorization for Network Access Server (NAS) Management
- RFC5904: RADIUS Attributes for IEEE 802.16 Privacy Key Management Version 1 (PKMv1) Protocol Support

Table 47-1: Standard RADIUS Attributes

Attribute ID and Name		Value Type/Pre-defined Values
1	User-Name	string
2	User-Password	string
3	CHAP-Password	octets (Hexadecimal string followed by 0x)
4	NAS-IP-Address	ipaddr (IPv4 address)
5	NAS-Port	Integer
6	Service-Type	Integer. Valid values are: ■ Administrative-User (6) ■ Authenticate-Only (8) ■ Authorize-Only (17) ■ Callback-Administrative (11) ■ Callback-Framed-User (4) ■ Callback-Login-User (3) ■ Callback-NAS-Prompt (9) ■ Call-Check (10) ■ Framed-Management (18) ■ Framed-User (2) ■ Login-User (1) ■ NAS-Prompt-User (7) ■ Outbound-User (5)
7	Framed-Protocol	Integer. Valid values are: ■ ARAP (3) ■ Gandalf-SLML (4) ■ PPP (1) ■ SLIP (2) ■ X.75-Synchronous (6) ■ Xylogics-IPX-SLIP (5)
8	Framed-IP-Address	ipaddr (IPv4 address)
9	Framed-IP-Netmask	ipaddr (IPv4 address)
10	Framed-Routing	integer. Valid values are: ■ Broadcast (1) ■ Broadcast-Listen (3) ■ Listen (2) ■ None (0)
11	Filter-Id	string
12	Framed-MTU	Integer
13	Framed-Compression	Integer. Valid values are: ■ IPX-Header-Compression (2) ■ None (0) ■ Stac-LZS (3) ■ Van-Jacobson-TCP-IP (1)
14	Login-IP-Host	IP Address

Table 47-1: Standard RADIUS Attributes(cont.)

Attribute ID and Name		Value Type/Pre-defined Values
15	Login-Service	Integer. Valid values are: ■ LAT (4) ■ PortMaster (3) ■ Rlogin (1) ■ TCP-Clear (2) ■ TCP-Clear-Quiet (8) ■ Telnet (0) ■ X25-PAD (5) ■ X25-T3POS (6)
16	Login-TCP-Port	Integer. Valid values are: ■ Rlogin (513) ■ Rsh (514) ■ Telnet (23)
18	Reply-Message	string
19	Callback-Number	string
20	Callback-Id	string
22	Framed-Route	string
23	Framed-IPX-Network	IP address
24	State	octets (Hexadecimal string followed by 0x)
25	Class	octets (Hexadecimal string followed by 0x)
26	Vendor-Specific	Use the Vendor-specific Attribute Name. For valid values, see "Vendor-Specific RADIUS Attributes" on page 47.17.
27	Session-Timeout	Integer
28	Idle-Timeout	Integer
29	Termination-Action	Integer. Valid values are: ■ Default (0) ■ RADIUS-Request (1)
30	Called-Station-Id	string
31	Calling-Station-Id	string
32	NAS-Identifier	string
33	Proxy-State	octets (Hexadecimal string followed by 0x)
34	Login-LAT-Service	string
35	Login-LAT-Node	string
36	Login-LAT-Group	octets (Hexadecimal string followed by 0x)
37	Framed-AppleTalk-Link	Integer
38	Framed-AppleTalk-Network	Integer

Table 47-1: Standard RADIUS Attributes(cont.)

Attribute ID and Name		Value Type/Pre-defined Values
39	Framed-AppleTalk-Zone	string
40	Acct-Status-Type	Integer. Valid values are: ■ Accounting-Off (8) ■ Accounting-On (7) ■ Alive (3) ■ Failed (15) ■ Interim-Update (3) ■ Start (1) ■ Stop (2) ■ Tunnel-Link-Reject (14) ■ Tunnel-Link-Start (12) ■ Tunnel-Link-Stop (13) ■ Tunnel-Reject (11) ■ Tunnel-Start (9) ■ Tunnel-Stop (10)
41	Acct-Delay-Time	Integer
42	Acct-Input-Octets	Integer
43	Acct-Output-Octets	Integer
44	Acct-Session-Id	string
45	Acct-Authentic	Integer. Valid values are: ■ Diameter (4) ■ Local (2) ■ RADIUS (1) ■ Remote (3)
46	Acct-Session-Time	Integer
47	Acct-Input-Packets	Integer
48	Acct-Output-Packets	Integer

Table 47-1: Standard RADIUS Attributes(cont.)

Attribute ID and Name		Value Type/Pre-defined Values
49	Acct-Terminate-Cause	Integer. Valid values are: ■ Admin-Reboot (7) ■ Admin-Reset (6) ■ Callback (16) ■ Host-Request (18) ■ Idle-Timeout (4) ■ Lost-Carrier (2) ■ Lost-Service (3) ■ NAS-Error (9) ■ NAS-Reboot (11) ■ NAS-Request (10) ■ Port-Disabled (22) ■ Port-Error (8) ■ Port-Preempted (13) ■ Port-Reinit (21) ■ Port-Suspended (14) ■ Port-Unneeded (12) ■ Reauthentication-Failure (20) ■ Service-Unavailable (15) ■ Session-Timeout (5) ■ Supplicant-Restart (19) ■ User-Error (17) ■ User-Request (1)
50	Acct-Multi-Session-Id	string
51	Acct-Link-Count	Integer
52	Acct-Input-Gigawords	Integer
53	Acct-Output-Gigawords	Integer
55	Event-Timestamp	date (Not supported)
56	Egress-VLANID	Integer
57	Ingress-Filters	Integer. Valid values are: ■ Disabled (2) ■ Enabled (1)
58	Egress-VLAN-Name	string
59	User-Priority-Table	octets (Hexadecimal string followed by 0x)
60	CHAP-Challenge	octets (Hexadecimal string followed by 0x)
61	NAS-Port-Type	Integer. Valid values are: ■ ADSL-CAP (12) ■ ADSL-DMT (13) ■ Async (0) ■ Cable (17) ■ Ethernet (15) ■ FDDI (21) ■ G.3-Fax (10) ■ HDLC-Clear-Channel (7)

Table 47-1: Standard RADIUS Attributes(cont.)

Attribute ID and Name		Value Type/Pre-defined Values
61	NAS-Port-Type (cont.)	Integer. Valid values are: ■ IDSL (14) ■ ISDN (2) ■ ISDN-V110 (4) ■ ISDN-V120 (3) ■ PIAFS (6) ■ PPPoA (30) ■ PPPoEoA (31) ■ PPPoEoE (32) ■ PPPoEoQinQ (34) ■ PPPoEoVLAN (33) ■ SDSL (11) ■ Sync (1) ■ Token-Ring (20) ■ Virtual (5) ■ Wireless-802.11 (19) ■ Wireless-Other (18) ■ X.25 (8) ■ X.75 (9) ■ xDSL (16)
62	Port-Limit	Integer
63	Login-LAT-Port	string
64	Tunnel-Type	Integer. Valid values are: ■ AH (6) ■ ATMP (4) ■ DVS (11) ■ ESP (9) ■ GRE (10) ■ IP (7) ■ IP-in-IP (12) ■ L2F (2) ■ L2TP (3) ■ MIN-IP (8) ■ PPTP (1) ■ VLAN (13) ■ VTP (5)

Table 47-1: Standard RADIUS Attributes(cont.)

Attribute ID and Name		Value Type/Pre-defined Values
65	Tunnel-Medium-Type	Integer. Valid values are: ■ Appletalk (12) ■ Banyan-Vines (14) ■ BBN-1822 (5) ■ DecNet-IV (13) ■ E.163 (7) ■ E.164 (8) ■ E.164-NSAP (15) ■ F.69 (9) ■ HDLC (4) ■ IEEE-802 (6) ■ IP (1) ■ IPv4 (1) ■ IPv6 (2) ■ IPX (11) ■ NSAP (3) ■ X.121 (10)
66	Tunnel-Client-Endpoint	string
67	Tunnel-Server-Endpoint	string
68	Acct-Tunnel-Connection	string
69	Tunnel-Password	string
70	ARAP-Password	octets (Hexadecimal string followed by 0x)
71	ARAP-Features	octets (Hexadecimal string followed by 0x)
72	ARAP-Zone-Access	Integer. Valid values are: ■ Default-Zone (1) ■ Zone-Filter-Exclusive (4) ■ Zone-Filter-Inclusive (2)
73	ARAP-Security	Integer
74	ARAP-Security-Data	string
75	Password-Retry	integer
76	Prompt	integer. Valid values are: ■ Echo (1) ■ No-Echo (0)
77	Connect-Info	string
78	Configuration-Token	string
79	EAP-Message	octets (Hexadecimal string followed by 0x)
80	Message-Authenticator	octets (Hexadecimal string followed by 0x)
81	Tunnel-Private-Group-Id	string
82	Tunnel-Assignment-Id	string
83	Tunnel-Preference	Integer

Table 47-1: Standard RADIUS Attributes(cont.)

Attribute ID and Name		Value Type/Pre-defined Values
84	ARAP-Challenge-Response	octets (Hexadecimal string followed by 0x)
85	Acct-Interim-Interval	Integer
86	Acct-Tunnel-Packets-Lost	Integer
87	NAS-Port-Id	string
88	Framed-Pool	string
89	Chargeable-User-Identity	string
90	Tunnel-Client-Auth-Id	string
91	Tunnel-Server-Auth-Id	string
92	NAS-Filter-Rule	string
95	NAS-IPv6-Address	ipv6addr (IPv6 address)
96	Framed-Interface-Id	ifid (Not supported)
97	Framed-IPv6-Prefix	ipv6prefix (Not supported)
98	Login-IPv6-Host	ipv6addr (IPv6 address)
99	Framed-IPv6-Route	string
100	Framed-IPv6-Pool	string
101	Error-Cause	Integer. Valid values are: ■ Administratively-Prohibited (501) ■ Invalid-Attribute-Value (407) ■ Invalid-EAP-Packet (202) ■ Invalid-Request (404) ■ Missing-Attribute (402) ■ Multiple-Session-Selection-Unsupported (508) ■ NAS-Identification-Mismatch (403) ■ Proxy-Processing-Error (505) ■ Proxy-Request-Not-Routable (502) ■ Request-Initiated (507) ■ Residual-Context-Removed (201) ■ Resources-Unavailable (506) ■ Session-Context-Not-Found (503) ■ Session-Context-Not-Removable (504) ■ Unsupported-Attribute (401) ■ Unsupported-Extension (406) ■ Unsupported-Service (405)
102	EAP-Key-Name	string
123	Delegated-IPv6-Prefix	ipv6prefix
126	Operator-Name	string
127	Location-Information	octets (Hexadecimal string followed by 0x)
128	Location-Data	octets (Hexadecimal string followed by 0x)

Table 47-1: Standard RADIUS Attributes(cont.)

Attribute ID and Name		Value Type/Pre-defined Values
129	Basic-Location-Policy-Rules	octets (Hexadecimal string followed by 0x)
130	Extended-Location-Policy-Rules	octets (Hexadecimal string followed by 0x)
131	Location-Capable	Integer. Valid values are: ■ Civix-Location (1) ■ Geo-Location (2) ■ NAS-Location (8) ■ Users-Location (4)
132	Requested-Location-Info	Integer. Valid values are: ■ Civix-Location (1) ■ Future-Requests (16) ■ Geo-Location (2) ■ NAS-Location (8) ■ None (32) ■ Users-Location (4)
133	Framed-Management	Integer. Valid values are: ■ FTP (4) ■ Netconf (3) ■ RCP (7) ■ SCP (8) ■ SFTP (6) ■ SNMP (1) ■ TFTP (5)
134	Management-Transport-Protection	Integer. Valid values are: ■ Integrity-Confidentiality-Protection (3) ■ Integrity-Protection (2) ■ No-Protection (1)
135	Management-Policy-Id	string
136	Management-Privilege-Level	Integer
137	PKM-SS-Cert	octets (Hexadecimal string followed by 0x)
138	PKM-CA-Cert	octets (Hexadecimal string followed by 0x)
139	PKM-Config-Settings	octets (Hexadecimal string followed by 0x)
140	PKM-Cryptosuite-List	octets (Hexadecimal string followed by 0x)
141	PKM-SAID	short
142	PKM-SA-Descriptor	octets (Hexadecimal string followed by 0x)
143	PKM-Auth-Key	octets (Hexadecimal string followed by 0x)

Table 47-2: Vendor-Specific RADIUS Attributes

Vendor-Specific Attribute Name	Value Type/Pre-defined Value
Actual-Data-Rate-Downstream	integer
Actual-Data-Rate-Upstream	integer
Actual-Interleaving-Delay-Downstream	integer
Actual-Interleaving-Delay-Upstream	integer
ADSL-Agent-Circuit-Id	string
ADSL-Agent-Remote-Id	string
Attainable-Data-Rate-Downstream	integer
Attainable-Data-Rate-Upstream	integer
call-id	string
Cisco-Abort-Cause	string
Cisco-Account-Info	string
Cisco-Assign-IP-Pool	integer
Cisco-AVPair	string
Cisco-Call-Filter	integer
Cisco-Call-Type	string
Cisco-Command-Code	string
Cisco-Control-Info	string
Cisco-Data-Filter	integer
Cisco-Data-Rate	integer

Table 47-2: Vendor-Specific RADIUS Attributes (cont.)

Vendor-Specific Attribute Name	Value Type/Pre-defined Value
Cisco-Disconnect-Cause	integer. Valid values are: ■ CLID-Authentication-Failure - 4 ■ Control-C-Detected - 27 ■ EXEC-Program-Destroyed - 28 ■ Exit-Raw-TCP - 24 ■ Exit-Telnet-Session - 22 ■ Failed-PPP-CHAP-Auth - 43 ■ Failed-PPP-LCP-Negotiation - 41 ■ Failed-PPP-PAP-Auth-Fail - 42 ■ Failed-PPP-Remote-Auth - 44 ■ Idle-Timeout - 21 ■ Invalid-Protocol - 120 ■ Lost-Carrier - 1 ■ No-Carrier - 0 ■ No-Detected-Result-Codes - 2 ■ No-Remote-IP-Addr - 23 ■ Password-Fail - 25 ■ PPP-Closed-Event- 46 ■ PPP-Remote-Terminate - 45 ■ Raw-TCP-Disabled - 26 ■ Session-End-Callback - 02 ■ Session-Failed-Security - 01 ■ Session-Timeout - 00 ■ Timeout-PPP-LCP - 40 ■ Unknown - 2 ■ User-Ends-Session - 20
Cisco-Email-Server-Ack-Flag	string
Cisco-Email-Server-Address	string
Cisco-Fax-Account-Id-Origin	string
Cisco-Fax-Auth-Status	string
Cisco-Fax-Connect-Speed	string
Cisco-Fax-Coverpage-Flag	string
Cisco-Fax-Dsn-Address	string
Cisco-Fax-Dsn-Flag	string
Cisco-Fax-Mdn-Address	string
Cisco-Fax-Mdn-Flag	string
Cisco-Fax-Modem-Time	string
Cisco-Fax-Msg-Id	string
Cisco-Fax-Pages	string
Cisco-Fax-Process-Abort-Flag	string
Cisco-Fax-Recipient-Count	string
Cisco-Gateway-Id	string

Table 47-2: Vendor-Specific RADIUS Attributes (cont.)

Vendor-Specific Attribute Name	Value Type/Pre-defined Value
Cisco-Idle-Limit	integer
Cisco-IP-Direct	integer
Cisco-IP-Pool-Definition	string
Cisco-Link-Compression	integer
Cisco-Maximum-Channels	integer
Cisco-Maximum-Time	integer
Cisco-Multilink-ID	integer
Cisco-NAS-Port	string
Cisco-Num-In-Multilink	integer
Cisco-Policy-Down	string
Cisco-Policy-Up	string
Cisco-Port-Used	string
Cisco-PPP-Async-Map	integer
Cisco-PPP-VJ-Slot-Comp	integer
Cisco-Pre-Input-Octets	integer
Cisco-Pre-Input-Packets	integer
Cisco-Pre-Output-Octets	integer
Cisco-Pre-Output-Packets	integer
Cisco-PreSession-Time	integer
Cisco-PW-Lifetime	integer
Cisco-Route-IP	integer
Cisco-Service-Info	string
Cisco-Subscriber-Password	string
Cisco-Target-Util	integer
Cisco-Xmit-Rate	integer
gw-final-xlated-cdn	string
gw-final-xlated-cgn	string
gw-rxd-cdn	string
gw-rxd-cgn	string
h323-billing-model	string
h323-call-origin	string

Table 47-2: Vendor-Specific RADIUS Attributes (cont.)

Vendor-Specific Attribute Name	Value Type/Pre-defined Value
h323-call-type	string
h323-conf-id	string
h323-connect-time	string
h323-credit-amount	string
h323-credit-time	string
h323-currency	string
h323-disconnect-cause	string
h323-disconnect-time	string
h323-gw-id	string
h323-incoming-conf-id	string
h323-preferred-lang	string
h323-prompt-id	string
h323-redirect-ip-address	string
h323-redirect-number	string
h323-remote-address	string
h323-return-code	string
h323-setup-time	string
h323-time-and-day	string
h323-voice-quality	string
incoming-req-uri	string
IWF-Session	octets
Maximum-Data-Rate-Downstream	integer
Maximum-Data-Rate-Upstream	integer
Maximum-Interleaving-Delay-Downstream	integer
Maximum-Interleaving-Delay-Upstream	integer
method	string
Minimum-Data-Rate-Downstream	integer
Minimum-Data-Rate-Downstream-Low-Power	integer
Minimum-Data-Rate-Upstream	integer
Minimum-Data-Rate-Upstream-Low-Power	integer

Table 47-2: Vendor-Specific RADIUS Attributes (cont.)

Vendor-Specific Attribute Name	Value Type/Pre-defined Value
MS-Acct-Auth-Type	integer. Valid values are: ■ CHAP - 2 ■ EAP - 5 ■ MS-CHAP-1 - 3 ■ MS-CHAP-2 - 4 ■ PAP - 1
MS-Acct-EAP-Type	integer. Valid values are: ■ Generic-Token-Card - 6 ■ MD5 - 4 ■ OTP - 5 ■ TLS -13
MS-AFW-Protection-Level	integer. Valid values are: ■ HECP-Response-Sign-And-Encrypt - 2 ■ HECP-Response-Sign-Only - 1
MS-AFW-Zone	integer. Valid values are: ■ MS-AFW-Zone-Boundary-Policy - 1 ■ MS-AFW-Zone-Protected-Policy - 3 ■ MS-AFW-Zone-Unprotected-Policy - 2
MS-ARAP-PW-Change-Reason	integer. Valid values are: ■ Admin-Requires-Password-Change - 3 ■ Expired-Password - 2 ■ Just-Change-Password - 1 ■ Password-Too-Short - 4
MS-BAP-Usage	integer. Valid values are: ■ Allowed - 1 ■ Not-Allowed - 0 ■ Required - 2
MS-CHAP2-CPW	octets
MS-CHAP2-Response	octets
MS-CHAP2-Success	octets
MS-CHAP-Challenge	octets
MS-CHAP-CPW-1	octets
MS-CHAP-CPW-2	octets
MS-CHAP-Domain	string
MS-CHAP-Error	string
MS-CHAP-LM-Enc-PW	octets
MS-CHAP-MPPE-Keys	octets
MS-CHAP-NT-Enc-PW	octets
MS-CHAP-Response	octets

Table 47-2: Vendor-Specific RADIUS Attributes (cont.)

Vendor-Specific Attribute Name	Value Type/Pre-defined Value
MS-Extended-Quarantine-State	integer. Valid values are: ■ Infected - 2 ■ No-Data - 4 ■ Transition - 1 ■ Unknown - 3
MS-Filter	octets
MS-HCAP-Location-Group-Name	string
MS-HCAP-User-Groups	string
MS-HCAP-User-Name	string
MS-Identity-Type	integer. Valid values are: ■ Ignore-User-Lookup-Failure - 2 ■ Machine-Health-Check - 1
MS-IPv4-Remediation-Servers	octets
MS-IPv6-Filter	octets
MS-IPv6-Remediation-Servers	octets
MS-Link-Drop-Time-Limit	integer
MS-Link-Utilization-Threshold	integer
MS-Machine-Name	string
MS-MPPE-Encryption-Policy	octets
MS-MPPE-Encryption-Type	octets
MS-MPPE-Encryption-Types	octets
MS-MPPE-Recv-Key	octets
MS-MPPE-Send-Key	octets
MS-Network-Access-Server-Type	integer. Valid values are: ■ DHCP-Server - 3 ■ HCAP-Server - 6 ■ HRA - 5 ■ Remote-Access-Server - 2 ■ Terminal-Server-Gateway - 1 ■ Unspecified - 0 ■ Wireless-Access-Point - 4
MS-New-ARAP-Password	octets
MS-Old-ARAP-Password	octets
MS-Primary-DNS-Server	ipaddr
MS-Primary-NBNS-Server	ipaddr
MS-Quarantine-Grace-Time	integer

Table 47-2: Vendor-Specific RADIUS Attributes (cont.)

Vendor-Specific Attribute Name	Value Type/Pre-defined Value
MS-Quarantine-IPFilter	octets
MS-Quarantine-Session-Timeout	integer
MS-Quarantine-SOH	octets
MS-Quarantine-State	integer. Valid values are: ■ Full-Access - 0 ■ Probation - 2 ■ Quarantine - 1
MS-Quarantine-User-Class	string
MS-RAS-Client-Name	string
MS-RAS-Client-Version	string
MS-RAS-Correlation	octets
MS-RAS-Vendor	integer
MS-RAS-Version	string
MS-RNAP-Not-Quarantine-Capable	integer. Valid values are: ■ SoH-Not-Sent - 1 ■ SoH-Sent - 0
MS-Secondary-DNS-Server	ipaddr
MS-Secondary-NBNS-Server	ipaddr
MS-Service-Class	string
MS-TSG-Device-Redirection	integer
MS-User-IPv4-Address	ipaddr
MS-User-IPv6-Address	ipv6addr
MS-User-Security-Identity	string
next-hop-dn	string
next-hop-ip	string
outgoing-req-uri	string
prev-hop-ip	string
prev-hop-via	string
release-source	string
remote-media-address	string
session-protocol	string
sip-conf-id	string

Table 47-2: Vendor-Specific RADIUS Attributes (cont.)

Vendor-Specific Attribute Name	Value Type/Pre-defined Value
sip-hdr	string
subscriber	string

Chapter 48: Local RADIUS Server Commands

Command List

This chapter provides an alphabetical reference for commands used to configure the local RADIUS server on the device. For more information, see Chapter 47, Local RADIUS Server Introduction and Configuration.

attribute

Use this command to define a RADIUS attribute for the local RADIUS server user group.

For a complete list of defined RADIUS attributes and values, see "Defined RADIUS attributes list" on page 47.8.

When used with the **help** parameter the **attribute** command displays a list of standard and vendor specific valid RADIUS attributes that are supported by the local RADIUS server.

If an attribute name is specified with the **help** parameter, then the **attribute** command displays a list of predefined attribute names. Note that you can only use the defined RADIUS attribute names and not define your own.

When used with the **value** parameter the **attribute** command configures RADIUS attributes to the user group. If the specified attribute is already defined then it is replaced with the new value.

Use the **no** variant of this command to delete an attribute from the local RADIUS server user group.

Syntax

```
attribute [<attribute-name>|<attribute-id>] help

attribute {<attribute-name>|<attribute-id>} <value>

no attribute {<attribute-name>|<attribute-id>}
```

Parameter	Description
`<attribute-name>`	RADIUS attribute name for standard attributes (see Table 47-1 on page 47.9) or Vendor-Specific attributes (see Table 47-2 on page 47.17).
`<attribute-id>`	RADIUS attribute numeric identifier for standard attributes (Table 47-1 on page 47.9).
`<value>`	RADIUS attribute value.
`help`	Display a list of available attribute types.

Default By default, no attributes are configured.

Mode RADIUS Server Group Configuration

Usage For the Standard attributes, the attribute may be specified using either the attribute name, or its numeric identifier. For example, command:

```
awplus(config-radsrv-group)# attribute acct-terminate-cause
                             help
```

will produce the same results as command:

```
awplus(config-radsrv-group)# attribute 49 help
```

In the same way, where the specific attribute has a pre-defined value, the parameter <*value*> may be substituted with the Value Name or with its numeric value, for example command:

```
awplus(config-radsrv-group)# attribute acct-terminate-cause
                             user-request
```

will produce the same results as command:

```
awplus(config-radsrv-group)# attribute 49 1
```

or command:

```
awplus(config-radsrv-group)# attribute acct-terminate-cause 1
```

Example To check a list of all available defined RADIUS attribute names, use the following commands:

```
                      awplus# configure terminal
              awplus(config)# radius-server local
       awplus(config-radsrv)# group Admin
 awplus(config-radsrv-group)# attribute help
```

A list of Vendor-specific Attributes displays after the list of defined Standard Attributes.

To get help for valid RADIUS attribute values for the attribute `Service-Type`, use the following commands:

```
                  awplus# configure terminal
          awplus(config)# radius-server local
   awplus(config-radsrv)# group Admin
awplus(config-radsrv-group)# attribute Service-Type help
```

```
Service-Type : integer (Integer number)
Pre-defined values :
    Administrative-User (6)
    Authenticate-Only (8)
    Authorize-Only (17)
    Callback-Administrative (11)
    Callback-Framed-User (4)
    Callback-Login-User (3)
    Callback-NAS-Prompt (9)
    Call-Check (10)
    Framed-User (2)
    Login-User (1)
    NAS-Prompt-User (7)
    Outbound-User (5)
```

To define the attribute name ‘Service-Type’ with Administrative User (6) to the RADIUS User Group ‘Admin’, use the following commands:

```
                  awplus# configure terminal
          awplus(config)# radius-server local
   awplus(config-radsrv)# group Admin
awplus(config-radsrv-group)# attribute Service-Type 6
```

To delete the attribute ‘Service-Type’ from the RADIUS User Group ‘Admin’, use the following commands:

```
                  awplus# configure terminal
          awplus(config)# radius-server local
   awplus(config-radsrv)# group Admin
awplus(config-radsrv-group)# no attribute Service-Type
```

Related Commands egress-vlan-id
egress-vlan-name

authentication

Use this command to enable the specified authentication methods on the local RADIUS server.

Use the **no** variant of this command to disable specified authentication methods on the local RADIUS server.

Syntax

```
authentication {mac|eapmd5|eaptls|peap}
```

```
no authentication {mac|eapmd5|eaptls|peap}
```

Parameter	Description
mac	Enable MAC authentication method.
eapmd5	Enable EAP-MD5 authentication method.
eaptls	Enable EAP-TLS authentication method.
peap	Enable EAP-PEAP authentication method.

Default All authentication methods are enabled by default.

Mode RADIUS Server Configuration

Example The following commands enable EAP-MD5 authentication methods on the local RADIUS server.

```
           awplus# configure terminal
   awplus(config)# radius-server local
awplus(config-radsrv)# authentication eapmd5
```

The following commands disable EAP-MD5 authentication methods on Local RADIUS server.

```
           awplus# configure terminal
   awplus(config)# radius-server local
awplus(config-radsrv)# no authentication eapmd5
```

Related Commands server enable
show radius local-server statistics

clear radius local-server statistics

Use this command to clear the statistics stored on the switch for the local RADIUS server.

Use this command without any parameters to clear all types of local RADIUS server statistics.

Syntax

```
clear radius local-server statistics [nas|server|user]
```

Parameter	Description
nas	Clear the NAS (Network Access Server) statistics on the switch. For example, clearing statistics stored for NAS server invalid passwords.
server	Clear the Local RADIUS Server statistics on the switch. For example, clearing Local RADIUS Servers statistics for all failed login attempts.
user	Clear the Local RADIUS Server user statistics. For example, clearing statistics stored for the number of successful user logins.

Mode Privileged Exec

Usage Refer to the sample output for the show radius local-server statistics for further information about the type of statistics each parameter option for this command clears. Both the **nas** and **server** parameters clear unknown username and invalid passwords statistics, while the `user` parameter clears the number of successful and failed logins for each local RADIUS server user.

Example To clear the NAS (Network Access Server) statistics stored on the switch, use the command:.

```
awplus# clear radius local-server statistics nas
```

To clear the local RADIUS server statistics stored on the switch, use the command:.

```
awplus# clear radius local-server statistics server
```

To clear the local RADIUS server user statistics stored on the switch, use the command:.

```
awplus# clear radius local-server statistics user
```

Related Commands show radius local-server statistics

copy fdb-radius-users (to file)

Use this command to create a set of local RADIUS server users from MAC addresses in the local FDB. A local RADIUS server user created using this command can be used for MAC authentication.

Syntax

```
copy fdb-radius-users {local-radius-user-db|flash|nvs|usb|debug|tftp|
   scp|<url>} [interface <port>] [vlan <vid>] [group <name>] [export-
   vlan [<group-name>]]
```

Parameter	Description
`local-radius-user-db`	Copy the local RADIUS server users created to the local RADIUS server.
`flash`	Copy the local RADIUS server users created to Flash memory.
`nvs`	Copy the local RADIUS server users created to NVS memory.
`usb`	Copy the local RADIUS server users created to a USB storage device.
`debug`	Copy the local RADIUS server users created to debug.
`tftp`	Copy the local RADIUS server users created to the TFTP destination.
`scp`	Copy the local RADIUS server users created to the SCP destination.
`<url>`	Copy the local RADIUS server users created to the specified URL.
`interface <port>`	Copy only MAC addresses learned on a specified switch port. Wildcards may be used when specifying an interface name. For example, when you specify interface port1.* then this command generates RADIUS server users for MAC addresses learned on stack 1.
`vlan <vid>`	Copy only MAC addresses learned on a specified VLAN.
`group <name>`	Assign a RADIUS group name to the local RADIUS server users created.
`export-vlan <group-name>`	Assign a RADIUS group name to the assigned export VLAN.

Mode Privileged Exec

Usage The local RADIUS server users created are written to a specified destination file in local RADIUS user CSV (Comma Separated Values) format. The local RADIUS server users can then be imported to a local RADIUS server using the copy local-radius-user-db (from file) command.

The name and password of the local RADIUS server users created use a MAC address, which can be used for MAC authentication.

This command does not copy a MAC address learned by the CPU or the management port.

This command can filter FDB entries by the interface name and the VLAN ID. When the interface name and the VLAN ID are specified, this command generates local RADIUS server users from only the MAC address learned on the specified interface and on the specified VLAN.

Examples To register the local RADIUS server users from the local FDB directly to the local RADIUS server, use the command:

```
awplus# copy fdb-radius-users local-radius-user-db
```

To register the local RADIUS server users from the interface `port1.0.1` to the local RADIUS server, use the command:

```
awplus# copy fdb-radius-users local-radius-user-db interface
        port1.0.1
```

To copy output generated as local RADIUS server user data from MAC addresses learned on `vlan10` on interface `port1.0.1` to the file `radius-user.csv`, use the command:

```
awplus# copy fdb-radius-users radius-user.csv interface
        port1.0.1 vlan10
```

Related Commands copy local-radius-user-db (to file)
copy local-radius-user-db (from file)

copy local-radius-user-db (from file)

Use this command to copy the Local RADIUS server user data from a file. The file, including the RADIUS user data in the file, must be in the CSV (Comma Separated Values) format.

You can select **add** or **replace** as the copy method. The **add** parameter option copies the contents of specified file to the local RADIUS server user database. If the same user exists then the old user is removed before adding a new user. The **replace** parameter option deletes all contents of the local RADIUS server user database before copying the contents of specified file.

Syntax

```
copy <source-url> local-radius-user-db [add|replace]
```

Parameter	Description
`<source-url>`	URL of the source file.
`add`	Add file contents to local RADIUS server user database.
`replace`	Replace current local RADIUS server user database with file contents.

Default When no copy method is specified with this command the **replace** option is applied.

Mode Privileged Exec

Examples To replace the current local RADIUS server user data to the contents of http://datahost/user.csv, use the following command:

```
awplus# copy http://datahost/user.csv local-radius-user-db
```

To add the contents of http://datahost/user.csv to the current local RADIUS server user database, use the following command:

```
awplus# copy http://datahost/user.csv local-radius-user-db add
```

Related commands

copy fdb-radius-users (to file)
copy local-radius-user-db (to file)

copy local-radius-user-db (to file)

Use this command to copy the local RADIUS server user data to a file. The output file produced is CSV (Comma Separated Values) format.

Syntax

```
copy local-radius-user-db {flash|nvs|usb|tftp|scp|<destination-url>}
```

Parameter	Description
`flash`	Copy to flash memory.
`nvs`	Copy to NVS memory.
`usb`	Copy to USB storage device.
`tftp`	Copy to TFTP destination.
`scp`	Copy to SCP destination.
`<destination-url>`	URL of the Destination file.

Mode Privileged Exec

Example Copy the current local RADIUS server user data to http://datahost/user.csv.

```
awplus# copy local-radius-user-db http://datahost/user.csv
```

Related Commands

copy fdb-radius-users (to file)
copy local-radius-user-db (from file)

crypto pki enroll local

Use this command to obtain a system certificate from the Local CA (Certificate Authority).

Use the **no** variant of this command to delete system certificates created by a Local CA (Certificate Authority).

Syntax

```
crypto pki enroll local
```

```
no crypto pki enroll local
```

Default The system certificate is not available until this command is issued.

Mode Global Configuration

Example The following command obtains the system certificate from the Local CA (Certificate Authority).

```
        awplus# configure terminal
awplus(config)# crypto pki enroll local
```

The following command deletes the system certificate created by the Local CA (Certificate Authority).

```
        awplus# configure terminal
awplus(config)# no crypto pki enroll local
```

Related Commands crypto pki trustpoint local
group

crypto pki enroll local local-radius-all-users

Use this command to create certificates for all users registered in the local RADIUS server. These certificates are created by the Local Certificate Authority (CA) on the switch.

Syntax

```
crypto pki enroll local local-radius-all-users
```

Default By default, there are no certificates for users in the local RADIUS server.

Mode Global Configuration

Example The following command obtains the local RADIUS server certificates for the user from the Local CA (Certificate Authority).

```
        awplus# configure terminal
awplus(config)# crypto pki enroll local local-radius-all-users
```

Related Commands

crypto pki trustpoint local
show crypto pki certificates

crypto pki enroll local user

Use this command to obtain a local user certificate from the Local CA (Certificate Authority).

Use the **no** variant of this command to delete user certificates created by the Local CA (Certificate Authority).

Syntax

```
crypto pki enroll local user <user-name>

no crypto pki enroll local user <user-name>
```

Parameter	Description
<user-name>	User name.

Default By default, there is no user certificate.

Mode Global Configuration

Example The following command obtains Tom's certificate from the Local CA (Certificate Authority).

```
        awplus# configure terminal
awplus(config)# crypto pki enroll local user Tom
```

The following command deletes Tom's certificates created by the Local CA (Certificate Authority):

```
        awplus# configure terminal
awplus(config)# no crypto pki enroll local user Tom
```

Related Commands crypto pki trustpoint local
show crypto pki certificates

crypto pki export local pem

Use this command to export the certificate associated with the Local CA to a PEM format file.

Syntax

```
crypto pki export local pem url <url>
```

Parameter	Description
<url>	URL string.

Mode Global Configuration

Example The following command exports the Local CA certificate to a PEM format file.

```
        awplus# configure terminal
awplus(config)# crypto pki export local pem url tftp://
                192.168.1.1/cacert.pem
```

Related Commands crypto pki enroll local

crypto pki export local pkcs12

Use this command to export a specified certificate to a PKCS12 format file.

This command cannot be used for exporting certificates for the local system.

Syntax

```
crypto pki export local pkcs12 <user-name> <destination-url>
```

Parameter	Description
<user-name>	User name.
<destination-url>	Destination URL string.

Mode Global Configuration

Example The following commands exports a certificate for a user named **client** to a PKCS12 format file.

```
        awplus# configure terminal
awplus(config)# crypto pki export local pkcs12 client tftp://
                192.168.1.1/cacert.pem
```

To export Tom's certificate to PKSC12 format file, use the commands:

```
        awplus# configure terminal
awplus(config)# crypto pki export local pksc12 Tom tftp://
                192.168.1.1/tom.pfx
```

Related Commands crypto pki enroll local

crypto pki trustpoint local

Use this command to declare the Local CA (Certificate Authority) as the trustpoint that the system uses. The ca-trustpoint configuration mode is available after this command is issued.

Use the **no** variant of this command to delete all information and certificates associated with Local CA as the trustpoint.

Syntax

```
crypto pki trustpoint local

no crypto pki trustpoint local
```

Default Local CA is not a trustpoint.

Mode Global Configuration

Examples Use the following commands to declare the Local CA as the trustpoint.

```
        awplus# configure terminal
awplus(config)# crypto pki trustpoint local
```

Use the following commands to delete all information and certificates associated with the Local CA.

```
        awplus# configure terminal
awplus(config)# no crypto pki trustpoint local
```

To create a client certificate for all users registered to the local RADIUS server, use the following commands:

```
       awplus(config)# crypto pki trustpoint local
awplus(ca-trust-point)# exit
       awplus(config)# crypto pki enroll local alternative
```

Related Commands

crypto pki enroll local
show crypto pki trustpoints

debug crypto

Use this command to enable Public Key Infrastructure (PKI) debugging. When PKI debugging is enabled, the PKI module starts generating diagnostic messages to the system log.

Use the **no** variant of this command to disable Public Key Infrastructure (PKI) debugging. When PKI debugging is disabled, the PKI module stops generating diagnostic messages to the system log.

Syntax

```
debug crypto pki
```

```
no debug crypto pki
```

Default PKI debugging is disabled by default

Mode Privileged Exec

Example To enable the PKI debugging facility, use the command:

```
awplus# debug crypto pki
```

To disable the PKI debugging facility, use the command:

```
awplus# no debug crypto pki
```

domain-style

Use this command to enable a specified domain style on the local RADIUS server. The local RADIUS server decodes the domain portion of a username login string when this command is enabled.

Use the **no** variant of this command to disable the specified domain style on the local RADIUS server.

Syntax

```
domain-style {suffix-atsign|ntdomain}
```

Parameter	Description
`suffix-atsign`	Enable at sign "@" delimited suffix style, i.e. "user@domain".
`ntdomain`	Enable NT domain style, i.e. "domain\user".

Default This feature is disabled by default.

Mode RADIUS Server

Usage When both domain styles are enabled, the first domain style configured has the highest priority. A username login string is matched against the first domain style enabled. Then, if the username login string is not decoded, it is matched against the second domain style enabled.

Example To enable NT domain style on the local RADIUS server, use the commands:

```
               awplus# configure terminal
       awplus(config)# radius-server local
awplus(config-radsrv)# domain-style ntdomain
```

To disable NT domain style on the local RADIUS server, use the commands:

```
               awplus# configure terminal
       awplus(config)# radius-server local
awplus(config-radsrv)# no domain-style ntdomain
```

Related Commands server enable

egress-vlan-id

Use this command to configure the standard RADIUS attribute 'Egress-VLANID (56)' for the local RADIUS Server user group.

Use the **no** variant of this command to remove the Egress-VLANID attribute from the local RADIUS server user group.

Syntax

```
egress-vlan-id <vid> [tagged|untagged]

no egress-vlan-id
```

Parameter	Description
`<vid>`	The VLAN identifier to be used for the Egress VLANID attribute, in the range 1 to 4094.
`tagged`	Set frames on the VLAN as tagged. This sets the tag indication field to indicate that all frames on this VLAN are tagged.
`untagged`	Set all frames on the VLAN as untagged. This sets the tag indication field to indicate that all frames on this VLAN are untagged.

Default By default, no Egress-VLANID attributes are configured.

Mode RADIUS Server Group Configuration

Usage When a Voice VLAN is configured for dynamic VLAN allocation (switchport voice vlan command on page 17.28), the RADIUS server must be configured to send the VLAN information when an IP phone is successfully authenticated. Use either the egress-vlan-id command or the egress-vlan-name command on page 48.20, and specify the **tagged** parameter.

Examples To set the 'Egress-VLANID' attribute for the NormalUsers local RADIUS server user group to VLAN identifier 200, with tagged frames, use the commands:

```
                awplus#  configure terminal
        awplus(config)#  radius-server local
 awplus(config-radsrv)#  group NormalUsers
awplus(config-radsrv-group)#  egress-vlan-id 200 tagged
```

To remove the 'Egress-VLANID' attribute for the NormalUsers local RADIUS server user group, use the commands:

```
                awplus#  configure terminal
        awplus(config)#  radius-server local
 awplus(config-radsrv)#  group NormalUsers
awplus(config-radsrv-group)#  no egress-vlan-id
```

Related Commands

attribute
egress-vlan-name
switchport voice vlan

egress-vlan-name

Use this command to configure the standard RADIUS attribute 'Egress-VLAN-Name (58)' for the local RADIUS server user group.

Use the **no** variant of this command to remove the Egress-VLAN-Name attribute from the local RADIUS server user group.

Syntax

```
egress-vlan-name <vlan-name> [tagged|untagged]
```

```
no egress-vlan-name
```

Parameter	Description
`<vlan-name>`	The VLAN name to be configured as the Egress-VLAN-Name attribute.
`tagged`	Set frames on the VLAN as tagged. This sets the tag indication field to indicate that all frames on this VLAN are tagged.
`untagged`	Set all frames on the VLAN as untagged. This sets the tag indication field to indicate that all frames on this VLAN are untagged.

Default By default, no Egress-VLAN-Name attributes are configured.

Mode RADIUS Server Group Configuration

Usage When a Voice VLAN is configured for dynamic VLAN allocation (switchport voice vlan command on page 17.28), the RADIUS server must be configured to send the VLAN information when an IP phone is successfully authenticated. Use either the egress-vlan-id command on page 48.19 or the egress-vlan-name command, and specify the **tagged** parameter.

Examples To configure the 'Egress-VLAN-Name' attribute for the RADIUS server user group NormalUsers with the VLAN name 'vlan2' and all frames on this VLAN tagged, use the commands:

```
                   awplus# configure terminal
           awplus(config)# radius-server local
    awplus(config-radsrv)# group NormalUsers
awplus(config-radsrv-group)# egress-vlan-name vlan2 tagged
```

To delete the 'Egress-VLAN-Name' attribute for the NormalUsers group, use the commands:

```
                   awplus# configure terminal
           awplus(config)# radius-server local
    awplus(config-radsrv)# group NormalUsers
awplus(config-radsrv-group)# no egress-vlan-name
```

Related Commands

attribute
egress-vlan-id
switchport voice vlan

group

Use this command to create a local RADIUS server user group, and enter local RADIUS Server User Group Configuration mode.

Use the **no** variant of this command to delete the local RADIUS server user group.

Syntax

```
group <user-group-name>

no group <user-group-name>
```

Parameter	Description
<user-group-name>	User group name string.

Mode RADIUS Server

Example The following command creates the user group NormalUsers.

```
           awplus# configure terminal
   awplus(config)# radius-server local
awplus(config-radsrv)# group NormalUsers
```

The following command deletes user group NormalUsers.

```
           awplus# configure terminal
   awplus(config)# radius-server local
awplus(config-radsrv)# no group NormalUsers
```

Related Commands

user (RADIUS server)
show radius local-server user
vlan (RADIUS server)

nas

This command adds a client device (the Network Access Server or the NAS) to the list of devices that are able to send authentication requests to the local RADIUS server. The NAS is identified by its IP address and a shared secret (also referred to as a shared key) must be defined that the NAS will use to establish its identity.

Use the **no** variant of this command to remove a NAS client from the list of devices that are allowed to send authentication requests to the local RADIUS server.

Syntax

```
nas <ip-address> key <nas-keystring>

no nas <ip-address>
```

Parameter	Description
<ip-address>	RADIUS NAS IP address.
<nas-keystring>	NAS shared keystring.

Mode RADIUS server

Example The following commands add the NAS with an IP address of `192.168.1.2` to the list of clients that may send authentication requests to the local RADIUS server. Note the shared key that this NAS will use to establish its identify is `NAS_PASSWORD`.

```
awplus# configure terminal
awplus(config)# radius-server local
awplus(config-radsrv)# nas 192.168.1.2 key NAS_PASSWORD
```

The following commands remove the NAS with an IP address of `192.168.1.2` from the list of clients that are allowed to send authentication requests to the local RADIUS server:

```
awplus# configure terminal
awplus(config)# radius-server local
awplus(config-radsrv)# no nas 192.168.1.2
```

Related Commands show radius local-server nas

radius-server local

Use this command to navigate to the Local RADIUS server configuration mode (`config-radsrv`) from the Global Configuration mode (`config`).

Syntax `radius-server local`

Mode Global Configuration

Example Local RADIUS Server commands are available from `config-radsrv` configuration mode. To change mode from User Exec mode to the Local RADIUS Server mode (config-radsrv), use the commands:

```
          awplus# configure terminal
  awplus(config)# radius-server local
awplus(config-radsrv)#
```

Output

```
awplus(config)#radius-server local
Creating Local CA repository.....OK
Enrolling Local System to local trustpoint..OK
awplus(config-radsrv)#
```

Related Commands server enable
show radius local-server group
show radius local-server nas
show radius local-server statistics
show radius local-server user

server auth-port

Use this command to change the UDP port number for local RADIUS server authentication.

Use the **no** variant of this command to reset the RADIUS server authentication port back to the default.

Syntax

```
server auth-port <1-65535>

no server auth-port
```

Parameter	Description
<1-65535>	UDP port number.

Default The default local RADIUS server UDP authentication port number is 1812.

Mode RADIUS Server

Example The following commands set the RADIUS server authentication port to 10000.

```
               awplus# configure terminal
       awplus(config)# radius-server local
awplus(config-radsrv)# server port 10000
```

The following commands reset the RADIUS server authentication port back to the default UDP port of 1812.

```
               awplus# configure terminal
       awplus(config)# radius-server local
awplus(config-radsrv)# no server port
```

Related Commands server enable
show radius local-server statistics

server enable

This command enables the local RADIUS server. The local RADIUS server feature is started immediately when this command is issued.

The **no** variant of this command disables local RADIUS server. When this command is issued, the local RADIUS server stops operating.

Syntax

```
server enable
```

```
no server enable
```

Default The local RADIUS server is disabled by default and must be enabled for use with this command.

Mode RADIUS Server

Examples To enable the local RADIUS server, use the following commands:

```
              awplus# configure terminal
      awplus(config)# radius-server local
awplus(config-radsrv)# server enable
```

To disable the local RADIUS server, use the command:

```
              awplus# configure terminal
      awplus(config)# radius-server local
awplus(config-radsrv)# no server enable
```

Related Commands server auth-port
show radius local-server statistics

show crypto pki certificates

Use this command to display certificate information for Local CA and Local System certificates.

For information on output options, see “Controlling “show” Command Output” on page 1.41.

Syntax

```
show crypto pki certificates [local-ca|local]
```

Parameter	Description
`local-ca`	Local CA certificate.
`local`	Local system certificate.

Mode User Exec and Privileged Exec

Example The following command displays Local CA (Certificate Authority) certificate information.

```
awplus# show crypto pki certificates local-ca
```

The following command displays Local System certificate information.

```
awplus# show crypto pki certificates local
```

The following command displays information for all Local CA and Local System certificates.

```
awplus# show crypto pki certificates
```

Output

Figure 48-1: Example output from the **show crypto pki certificates** command showing Local System and Local CA certificates

```
awplus#show crypto pki certificates
Certificate: Local System
    Data:
        Version: 3 (0x2)
        Serial Number: 4 (0x4)
        Signature Algorithm: sha1WithRSAEncryption
        Issuer: O=Allied-Telesis, CN=AlliedwarePlusCA
        Validity
            Not Before: Oct  8 07:50:55 2009 GMT
            Not After : Oct  6 07:50:55 2019 GMT
        Subject: O=Allied-Telesis, CN=Tom
Certificate: Local CA
    Data:
        Version: 3 (0x2)
        Serial Number: 0 (0x0)
        Signature Algorithm: sha1WithRSAEncryption
        Issuer: O=Allied-Telesis, CN=AlliedwarePlusCA
        Validity
            Not Before: Oct  8 07:55:55 2009 GMT
            Not After : Oct  6 07:55:55 2019 GMT
        Subject: O=Allied-Telesis, CN=Tom
```

Table 48-1: Parameters in the output of the **show crypto pki certificates** command

Parameter	Description
`Certificate`	Certificate name
`Version`	Protocol version
`Serial Number`	Serial number of the certificate
`Signature Algorithm`	Algorithm used for the certificate signature
`Issuer`	Subject of issuer creating the certificate
`Validity`	Validity period
`Subject`	Subject of the certificate

Related Commands crypto pki enroll local

show crypto pki certificates local-radius-all-users

Use this command to display certificate information for local RADIUS server users.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax

```
show crypto pki certificates local-radius-all-users
```

Mode User Exec and Privileged Exec

Example The following command displays information of all local RADIUS server user certificates.

```
awplus# show crypto pki certificates local-radius-all-users
```

Output

Figure 48-2: Example output from the **show crypto pki certificates local-radius-all-users** command

```
awplus#show crypto pki certificates local-radius-all-users
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 2 (0x2)
        Signature Algorithm: sha1WithRSAEncryption
        Issuer: O=Allied-Telesis, CN=AlliedwarePlusCA
        Validity
            Not Before: Oct  8 07:50:55 2009 GMT
            Not After : Oct  6 07:50:55 2019 GMT
        Subject: O=Allied-Telesis, CN=Tom
```

Table 48-2: Parameters in the output of the **show crypto pki certificates local-radius-all-users** command

Parameter	Description
`Certificate`	Certificate name
`Version`	Protocol version
`Serial Number`	Serial number of the certificate
`Signature Algorithm`	Algorithm used for the certificate signature
`Issuer`	Subject of issuer creating the certificate
`Validity`	Validity period
`Subject`	Subject of the certificate

Related Commands crypto pki enroll local local-radius-all-users

show crypto pki certificates user

Use this command to display certificate information for a specified local RADIUS server user.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax `show crypto pki certificates user [<user-name>]`

Parameter	Description
<user-name>	User name.

Mode User Exec and Privileged Exec

Example The following command displays Tom's certificate information.

```
awplus# show crypto pki certificates user Tom
```

Output

Figure 48-3: Example output from the **show crypto pki certificates user** command to show certificate information for user Tom

```
awplus#show crypto pki certificates user Tom
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 2 (0x2)
        Signature Algorithm: sha1WithRSAEncryption
        Issuer: O=Allied-Telesis, CN=AlliedwarePlusCA
        Validity
            Not Before: Oct  8 07:50:55 2009 GMT
            Not After : Oct  6 07:50:55 2019 GMT
        Subject: O=Allied-Telesis, CN=Tom
```

Table 48-3: Parameters in the output of the **show crypto pki certificates user** command

Parameter	Description
`Certificate`	Certificate name
`Version`	Protocol version
`Serial Number`	Serial number of the certificate
`Signature Algorithm`	Algorithm used for the certificate signature
`Issuer`	Subject of issuer creating the certificate
`Validity`	Validity period
`Subject`	Subject of the certificate

Related Commands crypto pki enroll local user

show crypto pki trustpoints

Use this command to display trustpoint information.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax

```
show crypto pki trustpoints
```

Mode User Exec and Privileged Exec

Example The following command displays trustpoint information.

```
awplus# show crypto pki trustpoint
```

Output

Figure 48-4: Example output from the **show crypto pki trustpoints** command

```
Trustpoint local:
Subject Name:
CN = AlliedwarePlusCA
o = Allied-Telesis
Serial Number:0C
```

Table 48-4: Parameters in the output of the **show crypto pki trustpoints** command

Parameter	Description
`Subject Name`	CA certificate subject.
`Serial Number`	Current serial number of CA.

Related Commands crypto pki enroll local

show radius local-server group

Use this command to display information about the local RADIUS server user group.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax

```
show radius local-server group [<user-group-name>]
```

Parameter	Description
<user-group-name>	User group name string.

Mode User Exec and Privileged Exec

Example The following command displays Local RADIUS server user group information.

```
awplus# show radius local-server group
```

Output

Figure 48-5: Example output from the **show radius local-server group** command

```
Group-Name          Vlan
----------------------------------------------------------------
NetworkOperators    ManagementNet
NormalUsers         CommonNet
```

Table 48-5: Parameters in the output of the **show radius local-server group** command

Parameter	Description
`Group-Name`	Group name.
`Vlan`	VLAN name assigned to the group.

Related Commands group

show radius local-server nas

Use this command to display information about NAS (Network Access Servers) registered to the local RADIUS server.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax `show radius local-server nas [<ip-address>]`

Parameter	Description
`<ip-address>`	Specify NAS IP address for show output.

Mode User Exec and Privileged Exec

Example The following command displays NAS information.

```
awplus# show radius local-server nas
```

Output

Figure 48-6: Example output from the **show radius local-server nas** command

```
NAS-Address    Shared-Key
----------------------------------------------------------
127.0.0.1      awplus-local-radius-server
```

Table 48-6: Parameters in the output of the **show radius local-server nas** command

Parameter	Description
`NAS-Address`	IP address of NAS.
`Shared-Key`	Shared key used for RADIUS connection.

Related Commands nas

show radius local-server statistics

Use this command to display statistics about the local RADIUS server.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax `show radius local-server statistics`

Mode User Exec and Privileged Exec

Usage Both unknown usernames and invalid passwords will display as failed logins in the show output.

Example The following command displays Local RADIUS server statistics.

```
awplus# show radius local-server statistics
```

Output

Figure 48-7: Example output from the **show radius local-server statistics** command

```
Server status  : Run (administrative status is enable)
Enabled methods: MAC EAP-MD5 EAP-TLS EAP-PEAP

Successes                :1   Unknown NAS               :0
Failed Logins            :0   Invalid packet from NAS   :0
Internal Error           :0   Unknown Error             :0

NAS : 127.0.0.1
Successes                :0   Shared key mismatch       :0
Failed Logins            :0   Unknown RADIUS message    :0
Unknown EAP message      :0   Unknown EAP auth type     :0
Corrupted packet         :0

NAS : 192.168.1.61
Successes                :0   Shared key mismatch       :0
Failed Logins            :0   Unknown RADIUS message    :0
Unknown EAP message      :0   Unknown EAP auth type     :0
Corrupted packet         :0

NAS : 192.168.1.63
Successes                :1   Shared key mismatch       :0
Failed Logins            :0   Unknown RADIUS message    :0
Unknown EAP message      :0   Unknown EAP auth type     :0
Corrupted packet         :0

NAS : 192.168.1.65
Successes                :0   Shared key mismatch       :0
Failed Logins            :0   Unknown RADIUS message    :0
Unknown EAP message      :0   Unknown EAP auth type     :0
Corrupted packet         :0

Username          Successes  Failures
a                 1          0
admin             0          0
```

Related Commands clear radius local-server statistics
radius-server local
server enable
server auth-port

show radius local-server user

Use this command to display information about the local RADIUS server user.

For information on output options, see “Controlling “show” Command Output” on page 1.41.

Syntax

```
show radius local-server user [<user-name>]

show radius local-server user <user-name> format csv
```

Parameter	Description
`<user-name>`	RADIUS user name. If no user name is specified, information for all users is displayed.
`format`	File format.
`csv`	Comma separated value format.

Mode User Exec and Privileged Exec

Usage You can send output from any show command, including the CSV format output from this command, to a file. See “Controlling “show” Command Output” on page 1.41.

Example The following command displays Local RADIUS server user information for user Tom.

```
awplus# show radius local-server user Tom
```

Figure 48-8: Example output from the show radius local-server user command

```
User-Name  Password   Group          Vlan
-----------------------------------------------------------------
Tom        abcd       NetworkOperators ManagementNet
```

The following command displays all Local RADIUS server information for all users.

```
awplus# show radius local-server user
```

The following command displays Local RADIUS server user information for TOM in CSV format.

```
awplus# show radius local-server user Tom format csv
```

Figure 48-9: Example output from the show radius local-server user csv command

```
true,"NetworkOperators","Tom",
"abcd",0,2099/01/
01,1,"","","ManagementNet",false,3600,false,0,"",false,"
```

Table 48-7: Parameters in the output from the **show radius local-server user** command

Parameter	Description
`User-Name`	User name.
`Password`	User password.
`Group`	Group name assigned to the user.
`Vlan`	VLAN name assigned to the user.

Related Commands group
user (RADIUS server)

user (RADIUS server)

Use this command to register a user to the local RADIUS server.

Use the **no** variant of this command to delete a user from the local RADIUS server.

Syntax

```
user <radius-user-name> [encrypted] password <user-password>
    [group <user-group>]
```

```
no user <radius-user-name>
```

Parameter	Description
`<radius-user-name>`	RADIUS user name. This can also be a MAC address in the IEEE standard format of `HH-HH-HH-HH-HH-HH` if you are configuring MAC authentication to use local RADIUS server.
`encrypted`	Specifies that the password is being entered in its encrypted form, so that it is not further encrypted. When creating a new user, enter the password in plaintext, and do not use the **encrypted** parameter. Use the **encrypted** parameter only when referring to a user that has previously been created. For instance, when adding an existing user from another RADIUS server, use the **encrypted** parameter, and enter the encrypted version of the password that appears in the output of **show** commands for the user.
`<user-password>`	User password. This can also be a MAC address in the IEEE standard format of `HH-HH-HH-HH-HH-HH` if you are configuring MAC authentication to use local RADIUS server.
`group`	Specify the group for the user.
`<user-group>`	User group name.

Mode RADIUS Server

Usage RADIUS user names cannot contain question mark (?), space ( ), or quote (“ ”) characters. RADIUS user names containing the below characters cannot use certificate authentication:

```
/ \ ` $ & ( ) * ; < > ` |
```

Certificates cannot be created and exported for RADIUS user names that contain the above characters. We advise you to avoid using these characters in RADIUS user names if you need to use certificate authentication, because you will not be able to create and export certificates.

You also can use the IEEE standard format hexadecimal notation (`HH-HH-HH-HH-HH-HH`) to specify a supplicant MAC address to configure the user name and user password parameters to use local RADIUS server for MAC Authentication. See the Sample MAC Authentication Configuration in Chapter 41, AAA Introduction and Configuration. See also the command **user 00-db-59-ab-70-37 password 00-db-59-ab-70-37** as shown in the command examples.

Examples The following commands add user `Tom` to the local RADIUS server and sets his password to `QwerSD`.

```
                awplus#  configure terminal
        awplus(config)#  radius-server local
  awplus(config-radsrv)#  user Tom password QwerSD
```

The following commands add user `Tom` to the local RADIUS server user group NormalUsers and sets his password `QwerSD`.

```
                awplus#  configure terminal
        awplus(config)#  radius-server local
  awplus(config-radsrv)#  user Tom password QwerSD group
                          NormalUsers
```

The following commands remove user Tom from the local RADIUS server:

```
                awplus#  configure terminal
        awplus(config)#  radius-server local
  awplus(config-radsrv)#  no user Tom
```

The following commands add the supplicant MAC address 00-d0-59-ab-70-37 to the local RADIUS server:

```
                awplus#  configure terminal
        awplus(config)#  radius-server local
  awplus(config-radsrv)#  user 00-db-59-ab-70-37 password 00-db-
                          59-ab-70-37
```

The following commands remove the supplicant MAC address 00-d0-59-ab-70-37 from the local RADIUS server:

```
                awplus#  configure terminal
        awplus(config)#  radius-server local
  awplus(config-radsrv)#  no user 00-db-59-ab-70-37
```

Related Commands group
show radius local-server user

vlan (RADIUS server)

Use this command to set the VLAN ID or name for the local RADIUS server user group. The VLAN information is used for authentication with the dynamic VLAN feature.

Use the **no** variant of this command to clear the VLAN ID or VLAN name for the local RADIUS server user group.

Syntax

```
vlan {<vid>|<vlan-name>}

no vlan
```

Parameter	Description
<vid>	VLAN ID.
<vlan-name>	VLAN name.

Default VLAN information is not set by default.

Mode RADIUS Server Group

Example The following commands set VLAN ID 200 to the group named NormalUsers:

```
                 awplus# configure terminal
         awplus(config)# radius-server local
  awplus(config-radsrv)# group NormalUsers
awplus(config-radsrv-group)# vlan 200
```

The following commands remove VLAN ID 200 from the group named NormalUsers:

```
                 awplus# configure terminal
         awplus(config)# radius-server local
  awplus(config-radsrv)# group NormalUsers
awplus(config-radsrv-group)# no vlan
```

Related Commands

group
show radius local-server user

Chapter 49: Secure Shell (SSH) Introduction

Introduction

This chapter describes how the Secure Shell protocol is implemented in the AlliedWare Plus^{TM} Operating System. It covers:

- Support for Secure Shell.
- Configuring your device as a Secure Shell server and client.
- Using Secure Shell to manage your device.

The AlliedWare Plus^{TM} OS supports SSH version 2 and SSH version 1.5, making it backwards compatible with SSH version 1.

Secure management is important in modern networks, as the ability to easily and effectively manage switches and routers, and the requirement for security, are two almost universal requirements. Protocols such as Telnet and rlogin allow you to manage devices remotely, but can have serious security problems, such as relying on reusable plaintext passwords that are vulnerable to wiretapping or password guessing. The Secure Shell (SSH) protocol is superior to these protocols by providing encrypted and strongly authenticated remote login sessions.

SSH provides sessions between a host running a SSH server and a machine with a SSH client. The AlliedWare Plus^{TM} OS includes both a SSH server and a SSH client to enable you to securely—with the benefit of cryptographic authentication and encryption—manage your devices over an insecure network:

- SSH replaces Telnet for remote terminal sessions; SSH is strongly authenticated and encrypted.
- Remote command execution allows you to send commands to a device securely and conveniently, without requiring a terminal session on the device.
- SSH allows you to connect to another host from your switch or router.

The AlliedWare Plus^{TM} OS supports Secure Copy (SCP) and SSH File Transfer Protocol (SFTP). Both these protocols allow you to securely copy files between your device and remote machines. SFTP provides additional features from SCP, such as allowing you to manipulate the remote files, and halt or resume file transfers without closing the session.

Secure Shell on the AlliedWare Plus OS

The AlliedWare Plus^{TM} OS implementation of SSH is compatible with the following RFCs and Internet Drafts:

- The Secure Shell (SSH) Protocol Architecture (RFC 4251)
- The Secure Shell (SSH) Authentication Protocol (RFC 4252)
- The Secure Shell (SSH) Transport Layer Protocol (RFC 4253)
- The Secure Shell (SSH) Connection Protocol (RFC 4254)
- The SSH (Secure Shell) Remote Login Protocol (draft-ylonen-ssh-protocol-00.txt)
- SSH File Transfer Protocol (draft-ietf-secsh-filexfer-13.txt)

Secure Shell supports the following features for both SSH version 2 and SSH version 1.5:

- Inbound SSH connections (server mode) and outbound SSH connections (client mode).
- File loading to and from remote machines using Secure Copy, using either the SSH client or SSH server mode.
- RSA public keys with lengths of 768–32768 bits, and DSA keys with lengths of 1024 bits. Keys are stored in a format compatible with other SSH implementations, and mechanisms are provided to copy keys to and from your device.
- Secure encryption, such as Triple DES and Blowfish.
- Remote non-interactive shell that allows arbitrary commands to be sent securely to your device, possibly automatically.
- Compression of Secure Shell traffic.
- Tunnelling of TCP/IP traffic.

Secure Shell supports the following features for SSH version 2 only:

- File loading from remote machines using SSH File Transfer Protocol (SFTP).
- A login banner on the SSH server, that displays when SSHv2 clients connect to the server.

Configuring the SSH Server

This section provides instructions on:

Creating a Host Key

The SSH server uses either an RSA or DSA host key to authenticate itself with SSH clients. This key must be configured before the SSH server can operate. If no host key exists, you cannot start the SSH server.

Once created, the host key is stored securely on the device. To generate a host key for the SSH server, use the command:

```
awplus(config)# crypto key generate hostkey {dsa|rsa|rsa1}
                [<768-32768>]
```

This command has two parameters for creating RSA keys. The **rsa** parameter creates a host key for SSH version 2 sessions only. To create a host key for SSH version 1 sessions, use the **rsa1** parameter.

To destroy a host key, use the command:

```
awplus(config)# crypto key destroy hostkey {dsa|rsa|rsa1}
```

To display a host key stored on your device, use the command:

```
awplus(config)# show crypto key hostkey [dsa|rsa|rsa1]
```

Enabling the Server

You must enable the SSH server before connections from SSH, SCP, and SFTP clients are accepted. When the SSH server is disabled it rejects connections from SSH clients. The SSH server is disabled by default on your device.

To enable the SSH server, use the command:

```
awplus(config)# service ssh [ip|ipv6]
```

To disable the SSH server, use the command:

```
awplus(config)# no service ssh [ip|ipv6]
```

When enabled, the SSH server allows SCP and SFTP sessions by default. To disable these services, use the commands:

```
awplus(config)# no ssh server scp
```

```
awplus(config)# no ssh server sftp
```

This allows you to reject SCP or SFTP file transfer requests, while still allowing Secure Shell connections. To re-enable SCP and SFTP services, use the command:

```
awplus(config)# ssh server scp
```

```
awplus(config)# ssh server sftp
```

Modifying the Server

To modify the SSH version that the server supports, or the TCP port that the server listens to for incoming sessions, use the command:

```
awplus(config)# ssh server {[v1v2|v2only]|<1-65535>}
```

The server listens on port 22 for incoming sessions, and supports both SSH version 2 and SSH version 1, by default.

To modify session and login timeouts on the SSH server, and the number of unauthenticated connections the server allows, use the command:

```
awplus(config)# ssh server {[session-timeout <0-3600>]
                [login-timeout <1-600>]
                [max-startups <1-128>]}
```

The SSH server waits 60 seconds for a client to authenticate itself, by default. You can alter this waiting time by using the **login-timeout** parameter. If the client is still not authenticated after the set timeout, then the SSH server disconnects the session.

The SSH server only allows only 10 unauthenticated SSH sessions at any point in time, by default. You can modify the number of unauthenticated sessions it allows, by using the **max-startups** parameter.

Once a client has authenticated, the SSH session does not time out, by default. Use the **session-timeout** parameter to set a **maximum time period the server waits before deciding that a session is inactive and terminating it**

For example, to set the session timeout to 600 seconds, the login timeout to 30 seconds, and the maximum number of concurrent unauthenticated sessions to 5, use the command:

```
awplus(config)# ssh server session-timeout 600 login-timeout
                30 max-startups 5
```

To remove the configured session timeout, login timeout, or maximum startups, use the command:

```
awplus(config)# no ssh server session-timeout login-timeout
                max-startups
```

Validating the Server Configuration

To validate the SSH server configuration, use the command:

```
awplus(config)# show running-config ssh
```

Adding SSH Users

The SSH server requires you to register SSH users. Users that are not registered cannot access the SSH server. Ensure first that you have defined the user in the Authorized User Database of your device. To add a new user, use the command:

```
awplus(config)# username USERNAME (privilege 1-15) password
                PASSWORD
```

To register a user with the SSH server, use the command:

```
awplus(config)# ssh server allow-users <username-pattern>
                [<hostname-pattern>]
```

Registered entries can contain just the username, or the username with some host details, such as an IP address range. Additionally you can specify a range of users or hostname details by using an asterisk to match any string of characters. For example, to allow any user from the IP range 192.168.1.1 to 192.168.1.255, use the command:

```
awplus(config)# ssh server allow-users * 192.168.1.*
```

To display the list of allowed users, use the command:

```
        awplus# show ssh server allow-users
```

To delete an entry from the list of allowed users, use the command:

```
awplus(config)# no ssh server allow-users <username-pattern>
                [<hostname-pattern>]
```

The SSH server also contains a list of denied users. The server checks all incoming sessions against this list and denies any matching session, regardless of whether the session matches an entry in the allowed users list. To add an entry to the list of denied users, use the command:

```
awplus(config)# ssh server deny-users <username-pattern>
                [<hostname-pattern>]
```

This allows you to deny specific users from a range of allowed users. For example, to deny a user with the IP address 192.168.1.12, use the command:

```
awplus(config)# ssh server deny-users * 192.168.1.12
```

To display the database of denied users, use the command:

```
        awplus# show ssh server deny-users
```

To delete a client from the database of denied users, use the command:

```
awplus(config)# no ssh server deny-users <username-pattern>
                [<hostname-pattern>]
```

Authenticating SSH Users

SSH users can use either their password or public key authentication to authenticate themselves with the SSH server. To use public key authentication, copy the user's public key file from their client device to the SSH server. To associate the key with a user, use the command:

```
awplus(config)# crypto key pubkey-chain userkey <username>
                [<filename>]
```

For example, to associate the file key.pub with the user "langley", use the command:

```
awplus(config)# crypto key pubkey-chain userkey langley
                key.pub
```

To add a key as text into the terminal for user "geoff", first enter the command:

```
awplus(config)# crypto key pubkey-chain userkey geoff
```

then paste or type the key in as text.

You can add multiple keys for the same user. To display the list of public keys associated with a user, use the command:

```
awplus(config)# show crypto key pubkey-chain userkey
                <username> [<1-65535>]
```

The **<1-65535>** parameter allows you to display an individual key.

To delete a key associated with a user from your device, use the command:

```
awplus(config)# no crypto key pubkey-chain userkey
                <username> <1-65535>
```

Adding a Login Banner

You can add a login banner to the SSH server for sessions with SSH version 2 clients. The server displays the banner to clients before the login prompt. To set the login banner's message, use the command:

```
awplus(config)# banner login
```

then enter your message and use Ctrl+D to finish.

To view the configured login banner, use the command:

```
        awplus# show banner login
```

To remove the configured message for the login banner, use the command:

```
awplus(config)# no banner login
```

Monitoring the Server and Managing Sessions

To display the current status of the SSH server, use the command:

```
awplus# show ssh server
```

To display the current status of SSH sessions on your device, use the command:

```
awplus# show ssh
```

Note that this displays both SSH server and SSH client sessions that your Allied Telesis device is running. Use this command to view the unique identification number assigned to each incoming or outgoing SSH session. You need the ID number when terminating a specific session from your device.

To terminate a session, or all sessions, use the command:

```
awplus# clear ssh {<1-65535>|all}
```

Debugging the Server

Information which may be useful for troubleshooting the SSH server is available using the SSH debugging function. You can enable server debugging while the SSH server is functioning. Use the command:

```
awplus# debug ssh server [brief|full]
```

To disable SSH server debugging, use the command:

```
awplus# no debug ssh server
```

Configuring the SSH Client

This section provides instructions on:

- Modifying the Client
- Adding SSH Servers
- Authenticating with a Server
- Connecting to a Server and Running Commands
- Copying files to and from the Server
- Debugging the Client

Modifying the Client

You can configure a selection of variables when using the SSH client. Note that the following configuration commands apply only to client sessions initiated after the command. The configured settings are not saved; after you have logged out from the SSH client, the client returns to using the default settings. Use the command:

```
awplus(config)# ssh client {port <1-65535>|version {1|2}|
                session-timeout <0-3600>|connect-timeout
                <1-600>}
```

The SSH client uses TCP port 22, by default. You can change the TCP port for the remote SSH server by using the **port** parameter.

The client supports both SSH version 1 and version 2 sessions, by default. To change the SSH client to only use a specific SSH version for sessions, for example SSH version 1, use the **version** parameter.

The client terminates sessions that are not established after 30 seconds, by default. You can change this time period by using the session-timeout parameter.

Once the client has authenticated with a server, the client does not time out the SSH session, by default. Use the **session-timeout** parameter to set a maximum time period the client waits before deciding that a session is inactive and terminating the session.

To modify the SSH client so that it uses port 2000 for sessions, and supports only SSH version 1 connections, use the command:

```
awplus(config# ssh client port 2000 version 1
```

To modify the SSH client so that unestablished sessions time out after 60 seconds, and inactive sessions time out after 100 seconds, use the command:

```
awplus(config)# ssh client session-timeout 100 connect-timeout
                100
```

To remove the configured port, SSH version, session timeout, and connection timeout settings, use the command:

```
awplus(config)# no ssh client port version session-timeout
                connect-timeout
```

Adding SSH Servers

SSH servers identify themselves using a host key (see "Creating a Host Key" on page 49.4). Before the SSH client establishes a session with a SSH server, it confirms that the host key sent by the server matches its database entry for the server. If the database does not contains a host key for the server, then the SSH client requires you to confirm that the host key sent from the server is correct.

To add an SSH server to the client's database, use the command:

```
awplus# crypto key pubkey-chain knownhosts [ip|ipv6]
        <hostname> [rsa|dsa|rsa1]
```

To display the SSH servers in the client's database, use the command:

```
awplus# show crypto key pubkey-chain knownhosts
        [<1-65535>]
```

To remove an entry in the database, use the command:

```
awplus# no crypto key pubkey-chain knownhosts <1-65535>
```

Authenticating with a Server

You can authenticate your session with a server by either using a password, or using RSA or DSA public key authentication. To use public key authentication, you must generate a pair of keys, one private and one public, and copy the public key onto the SSH server.

To generate an RSA or DSA set of private and public keys for an SSH user, use the command:

```
awplus(config)# crypto key generate userkey <username> {dsa|
                rsa|rsa1} [<768-32768>]
```

You can generate one key of each encryption type per user on your client. When authenticating with an SSH server that supports SSH version 1 only, you must use a key generated by the **rsa1** parameter.

To copy the public key onto the SSH server, you must display the key onscreen. To display the public key associated with a user, use the command:

```
awplus# show crypto key userkey <username> [dsa|rsa|
        rsa1]
```

To display the public keys set for other users, you must specify their username. Only users with the highest privilege setting can use this command to view the keys of other users.

To delete a public and private pair of keys associated with a user, use the command:

```
awplus(config)# crypto key destroy userkey <username> {dsa|rsa|
                rsa1}
```

Connecting to a Server and Running Commands

To connect to a remote SSH server and execute a command, use the command:

```
awplus# ssh [ip|ipv6][{[user <username>]|[port
        <1-65535>]|[version {1|2]}}] <hostname>
        [<line>]
```

By default, the SSH client attempts to use SSH version 2 with the SSH server. If this fails, the client uses SSH version 1.

For example, to connect to the SSH server at 192.168.1.2 as user "john", and execute the command "show sys", use the command:

```
awplus# ssh user john 192.168.1.2 "show sys"
```

Copying files to and from the Server

You can use either the SCP or SFTP client to transfer files from a remote SSH server. Use the command:

```
awplus# copy <source-url> <destination-url>
```

For example, to use SFTP to load a file from the SSH server 192.168.1.2, onto the flash memory of your device, use the command:

```
awplus# copy sftp://192.168.1.2/key.pub flash
```

To upload files to the SSH server, you must use SCP. For example, to upload the file bobskey.pub as the user "bob", use the command:

```
awplus# copy flash:/bobskey.pub scp://bob@192.168.1.2
```

For more information see Chapter 6, Creating and Managing Files.

Debugging the Client

Information which may be useful for troubleshooting the SSH client is available using the SSH debugging function. You can enable client debugging while the SSH client is functioning. Use the command:

```
awplus# debug ssh client [brief|full]
```

To disable SSH client debugging, use the command:

```
awplus# no debug ssh client
```

Chapter 50: Secure Shell (SSH) Configuration

SSH Server Configuration Example

This chapter provides a Secure Shell server configuration example. For more information about the SSH server, see Chapter 49, Secure Shell (SSH) Introduction. For detailed information about the commands used to configure the SSH server, see Chapter 51, Secure Shell (SSH) Commands.

The following example configures a SSH server where:

- the SSH server uses RSA encryption
- the SSH server is compatible with both SSH version 1 and version 2 clients
- three SSH users are configured: Manager, Colin and Asuka. "Manager" can connect from only a defined range of hosts, while "Colin" and "asuka" can SSH from all hosts
- the SSH users use RSA private and public key authentication

This example shows how to create RSA encryption keys, configure the Secure Shell server, and register users to make Secure Shell connections to your device.

Step 1: Login as a highest Privileged User.

To create the keys and add users, you must login as a privileged user.

Step 2: Create encryption keys.

Two RSA private keys are required before enabling the Secure Shell server for each type of SSH version. Use the commands:

```
        awplus#  configure terminal
awplus(config)#  crypto key generate hostkey rsa
awplus(config)#  crypto key generate hostkey rsa1
awplus(config)#  exit
```

To verify the key creation, use the command:

```
        awplus#  show crypto key hostkey
```

Step 3: Enable the Secure Shell server.

Enable Secure Shell on the device using the commands:

```
        awplus#  configure terminal
awplus(config)#  service ssh
```

Modify the SSH server settings as desired. For example, to set the login-timeout to 60, and the session-timeout to 3600, use the commands:

```
awplus(config)#  ssh server session-timeout 3600 login-timeout 60
```

To verify the server configuration, use the command:

```
        awplus#  show ssh
```

Step 4: Create SSH users.

In order to connect and execute commands, you must register users in the SSH user database, and in the User Authentication Database of the device.

To create the users `Colin` and `Asuka` in the User Authentication Database, use the commands:

```
        awplus# configure terminal
awplus(config)# username colin privilege 15 password secret
awplus(config)# username asuka privilege 15 password
                very-secret
```

To register `Colin` and `Asuka` as SSH clients, use the commands:

```
awplus(config)# ssh server allow-users colin
awplus(config)# ssh server allow-users asuka
```

To register “manager” as an SSH client so that can only connect from the IP address 192.168.1.1, use the command:

```
awplus(config)# ssh server allow-users manager 192.168.1.1
```

Step 5: Set up Authentication.

SSH users cannot connect unless the server can authenticate them. There are two ways to authenticate an SSH session: password authentication, and RSA or DSA private/public key authentication. When using password authentication, the user must supply their User Authentication Database password.

To use private/public key authentication, copy the public keys for each user onto the device. To copy the files onto flash from the key directory of an attached TFTP server, use the command:

```
        awplus# copy tftp://key/colin.pub flash:/colin.pub
        awplus# copy tftp://key/asuka.pub flash:/asuka.pub
```

To associate the key file with each user, use the command:

```
        awplus# configure terminal
awplus(config)# cryto key pubkey-chain userkey colin colin.pub
awplus(config)# cryto key pubkey-chain userkey asuka asuka.pub
awplus(config)# cryto key pubkey-chain userkey manager
                manager.pub
```

Chapter 51: Secure Shell (SSH) Commands

Introduction

This chapter provides an alphabetical reference for commands used to configure Secure Shell (SSH). For more information, see Chapter 49, Secure Shell (SSH) Introduction, and Chapter 50, Secure Shell (SSH) Configuration.

Command List

banner login (SSH)

This command configures a login banner on the SSH server. This displays a message on the remote terminal of the SSH client before the login prompt. SSH client version 1 does not support this banner.

To add a banner, first enter the command **banner login**, and hit [Enter]. Write your message. You can use any character and spaces. Use Ctrl+D at the end of your message to save the text and re-enter the normal command line mode.

The banner message is preserved if the device restarts.

The **no** variant of this command deletes the login banner from the device.

Syntax

```
banner login
no banner login
```

Default No banner is defined by default.

Mode Global Configuration

Examples To set a login banner message, use the commands:

```
        awplus# configure terminal
 awplus(config)# banner login

Type CNTL/D to finish.
... banner message comes here ...
^D
awplus(config)#
```

and enter the message. Use Ctrl+D to finish.

To remove the login banner message, use the commands:

```
        awplus# configure terminal
 awplus(config)# no banner login
```

Related Commands show banner login

clear ssh

This command deletes Secure Shell sessions currently active on the device. This includes both incoming and outgoing sessions. The deleted sessions are closed. You can only delete an SSH session if you are a system manager or the user who initiated the session. If **all** is specified then all active SSH sessions are deleted.

Syntax

```
clear ssh {<1-65535>|all}
```

Parameters	Description
<1-65535>	Specify a session ID in the range 1 to 65535 to delete a specific session.
`all`	Delete all SSH sessions.

Mode Privileged Exec

Examples To stop the current SSH session 123, use the command:

```
awplus# clear ssh 123
```

To stop all SSH sessions active on the device, use the command:

```
awplus# clear ssh all
```

Related Commands service ssh
ssh

crypto key destroy hostkey

This command deletes the existing public and private keys of the SSH server. Note that for an SSH server to operate it needs at least one set of hostkeys configured before an SSH server is started.

Syntax

```
crypto key destroy hostkey {dsa|rsa|rsa1}
```

Parameters	Description
`dsa`	Deletes the existing DSA public and private keys.
`rsa`	Deletes the existing RSA public and private keys configured for SSH version 2 connections.
`rsa1`	Deletes the existing RSA public and private keys configured for SSH version 1 connections.

Mode Global Configuration

Example To destroy the RSA host key used for SSH version 2 connections, use the commands:

```
        awplus# configure terminal
awplus(config)# crypto key destroy hostkey rsa
```

Related Commands crypto key generate hostkey
service ssh

crypto key destroy userkey

This command destroys the existing public and private keys of an SSH user configured on the device.

Syntax

```
crypto key destroy userkey <username> {dsa|rsa|rsa1}
```

Parameters	Description
`<username>`	Name of the user whose userkey you are destroying. The username must begin with a letter. Valid characters are all numbers, letters, and the underscore, hyphen and full stop symbols.
`dsa`	Deletes the existing DSA userkey.
`rsa`	Deletes the existing RSA userkey configured for SSH version 2 connections.
`rsa1`	Deletes the existing RSA userkey for SSH version 1 connections.

Mode Global Configuration

Example To destroy the RSA user key for the SSH user `remoteuser`, use the commands:

```
        awplus# configure terminal
awplus(config)# crypto key destroy userkey remoteuser rsa
```

Related Commands

crypto key generate hostkey
show ssh
show crypto key hostkey

crypto key generate hostkey

This command generates public and private keys for the SSH server using either an RSA or DSA cryptography algorithm. You must define a host key before enabling the SSH server. Start SSH server using the **service ssh** command. If a host key exists with the same cryptography algorithm, this command replaces the old host key with the new key.

This command is not saved in the device configuration. However, the device saves the keys generated by this command in the non-volatile memory.

Syntax

```
crypto key generate hostkey {dsa|rsa|rsa1} [<768-32768>]
```

Parameters	Description
`dsa`	Creates a DSA hostkey. Both SSH version 1 and 2 connections can use the DSA hostkey.
`rsa`	Creates an RSA hostkey for SSH version 2 connections.
`rsa1`	Creates an RSA hostkey for SSH version 1 connections.
`<768-32768>`	The length in bits of the generated key. The default is 1024 bits.

Default 1024 bits is the default key length. The DSA algorithm supports 1024 bits.

Mode Global Configuration

Examples To generate an RSA host key for SSH version 2 connections that is 2048 bits in length, use the commands:

```
        awplus# configure terminal
awplus(config)# crypto key generate hostkey rsa 2048
```

To generate a DSA host key, use the commands:

```
        awplus# configure terminal
awplus(config)# crypto key generate dsa
```

Related Commands

crypto key destroy hostkey
service ssh
show crypto key hostkey

crypto key generate userkey

This command generates public and private keys for an SSH user using either an RSA or DSA cryptography algorithm. To use public key authentication, copy the public key of the user onto the remote SSH server.

This command is not saved in the device configuration. However, the device saves the keys generated by this command in the non-volatile memory.

Syntax `crypto key generate userkey <username> {dsa|rsa|rsa1} [<768-32768>]`

Parameters	Description
<username>	Name of the user that the user key is generated for. The username must begin with a letter. Valid characters are all numbers, letters, and the underscore, hyphen and full stop symbols.
`dsa`	Creates a DSA userkey. Both SSH version 1 and 2 connections can use a key created with this command.
`rsa`	Creates an RSA userkey for SSH version 2 connections.
`rsa1`	Creates an RSA userkey for SSH version 1 connections.
<768-32768>	The length in bits of the generated key. The DSA algorithm supports only 1024 bits. Default: 1024.

Mode Global Configuration

Examples To generate a 2048-bits RSA user key for SSH version 2 connections for the user bob, use the commands:

```
        awplus# configure terminal
awplus(config)# crypto key generate userkey bob rsa 2048
```

To generate a DSA user key for the user lapo, use the commands:

```
        awplus# configure terminal
awplus(config)# crypto key generate userkey lapo dsa
```

Related Commands crypto key destroy userkey
show crypto key userkey

crypto key pubkey-chain knownhosts

This command adds a public key of the specified SSH server to the known host database on your switch. The SSH client on your switch uses this public key to verify the remote SSH server.

The key is retrieved from the server. Before adding a key to this database, check that the key sent to you is correct.

If the server's key changes, or if your SSH client does not have the public key of the remote SSH server, then your SSH client will inform you that the public key of the server is unknown or altered.

The **no** variant of this command deletes the public key of the specified SSH server from the known host database on your device.

Syntax

```
crypto key pubkey-chain knownhosts [ip|ipv6] <hostname> [rsa|dsa|
    rsa1]
```

```
no crypto key pubkey-chain knownhosts <1-65535>
```

Parameter	Description
`ip`	Keyword used prior to specifying an IPv4 address
`ipv6`	Keyword used prior to specifying an IPv6 address
`<hostname>`	IPv4/IPv6 address or hostname of a remote server in the format `a.b.c.d` for an IPv4 address, or in the format `x:x::x:x` for an IPv6 address
`rsa`	Specify the RSA public key of the server to be added to the known host database.
`dsa`	Specify the DSA public key of the server to be added to the known host database.
`rsa1`	Specify the SSHv1 public key of the server to be added to the know host database.
`<1-65535>`	Specify a key identifier when removing a key using the **no** parameter.

Default If no cryptography algorithm is specified, then **rsa** is used as the default cryptography algorithm.

Mode Privilege Exec

Usage This command adds a public key of the specified SSH server to the known host database on the switch. The key is retrieved from the server. The remote SSH server is verified by using this public key. The user is requested to check the key is correct before adding it to the database.

If the remote server's host key is changed, or if the device does not have the public key of the remote server, then SSH clients will inform the user that the public key of the server is altered or unknown.

Examples To add the RSA host key of the remote SSH host IPv4 address `192.0.2.11` to the known host database, use the command:

```
awplus# crypto key pubkey-chain knownhosts 192.0.2.11
```

To delete the second entry in the known host database, use the command:

```
awplus# no crypto key pubkey-chain knownhosts 2
```

Validation Commands

show crypto key pubkey-chain knownhosts

crypto key pubkey-chain userkey

This command adds a public key for an SSH user on the SSH server. This allows the SSH server to support public key authentication for the SSH user. When configured, the SSH user can access the SSH server without providing a password from the remote host.

The **no** variant of this command removes a public key for the specified SSH user that has been added to the public key chain. When a SSH user's public key is removed, the SSH user can no longer login using public key authentication.

Syntax

```
crypto key pubkey-chain userkey <username> [<filename>]

no crypto key pubkey-chain userkey <username> <1-65535>
```

Parameters	Description
<username>	Name of the user that the SSH server associates the key with. The username must begin with a letter. Valid characters are all numbers, letters, and the underscore, hyphen and full stop symbols. Default: no default
<filename>	Filename of a key saved in flash. Valid characters are any printable character. You can add a key as a hexadecimal string directly into the terminal if you do not specify a filename.
<1-65535>	The key ID number of the user's key. Specify the key ID to delete a key.

Mode Global Configuration

Usage You should import the public key file from the client node. The device can read the data from a file on the flash or user terminal.

Or you can add a key as text into the terminal. To add a key as text into the terminal, first enter the command **crypto key pubkey-chain userkey <*username*>**, and hit [Enter]. Enter the key as text. Note that the key you enter as text must be a valid SSH RSA key, not random ASCII text. Use [Ctrl]+D after entering it to save the text and re-enter the normal command line mode.

Note you can generate a valid SSH RSA key on the switch first using the **crypto key generate host rsa** command. View the SSH RSA key generated on the switch using the **show crypto hostkey rsa** command. Copy and paste the displayed SSH RSA key after entering the **crypto key pubkey-chain userkey <*username*>** command. Use [Ctrl]+D after entering it to save it.

Examples To generate a valid SSH RSA key on the switch and add the key, use the following commands:

```
        awplus# configure terminal

awplus(config)# crypto key generate host rsa

awplus(config)# exit

        awplus# show crypto key hostkey rsa

                AAAAB3NzaC1yc2EAAAABIwAAAIEAr1s7SokW5aW2fcOw1TStpb9J
                20bWluhnUC768EoWhyPW6FZ2t5360O5M29EpKBmGqlkQaz5V0mU9
                IQe66+5YyD4UxOKSDtTI+7jtjDcoGWHb2u4sFwRpXwJZcgYrXW16
                +6NvNbk+h+c/pqGDijj4SvfZZfeITzvvyZW4/I4pbN8=


        awplus# configure terminal

awplus(config)# crypto key pubkey-chain userkey joe
                Type CNTL/D to finish:

                AAAAB3NzaC1yc2EAAAABIwAAAIEAr1s7SokW5aW2fcOw1TStpb9J
                20bWluhnUC768EoWhyPW6FZ2t5360O5M29EpKBmGqlkQaz5V0mU9
                IQe66+5YyD4UxOKSDtTI+7jtjDcoGWHb2u4sFwRpXwJZcgYrXW16
                +6NvNbk+h+c/pqGDijj4SvfZZfeITzvvyZW4/I4pbN8=

                control-D

awplus(config)#
```

To add a public key for the user `graydon` from the file `key.pub`, use the commands:

```
        awplus# configure terminal

awplus(config)# crypto key pubkey-chain userkey graydon key.pub
```

To add a public key for the user `tamara` from the terminal, use the commands:

```
        awplus# configure terminal

awplus(config)# crypto key pubkey-chain userkey tamara
```

and enter the key. Use Ctrl+D to finish.

To remove the first key entry from the public key chain of the user `john`, use the commands:

```
        awplus# configure terminal

awplus(config)# no crypto key pubkey-chain userkey john 1
```

Related Commands show crypto key pubkey-chain userkey

debug ssh client

This command enables the SSH client debugging facility. When enabled, any SSH, SCP and SFTP client sessions send diagnostic messages to the login terminal.

The **no** variant of this command disables the SSH client debugging facility. This stops the SSH client from generating diagnostic debugging message.

Syntax

```
debug ssh client [brief|full]
```

```
no debug ssh client
```

Parameter	Description
`brief`	Enables brief debug mode.
`full`	Enables full debug mode.

Default SSH client debugging is disabled by default.

Mode Privileged Exec and Global Configuration

Examples To start SSH client debugging, use the command:

```
awplus# debug ssh client
```

To start SSH client debugging with extended output, use the command:

```
awplus# debug ssh client full
```

To disable SSH client debugging, use the command:

```
awplus# no debug ssh client
```

Related Commands

debug ssh server
show ssh client
undebug ssh client

debug ssh server

This command enables the SSH server debugging facility. When enabled, the SSH server sends diagnostic messages to the system log. To display the debugging messages on the terminal, use the **terminal monitor** command.

The **no** variant of this command disables the SSH server debugging facility. This stops the SSH server from generating diagnostic debugging messages.

Syntax

```
debug ssh server [brief|full]
```

```
no debug ssh server
```

Parameter	Description
`brief`	Enables brief debug mode.
`full`	Enables full debug mode.

Default SSH server debugging is disabled by default.

Mode Privileged Exec and Global Configuration

Examples To start SSH server debugging, use the command:

```
awplus# debug ssh server
```

To start SSH server debugging with extended output, use the command:

```
awplus# debug ssh server full
```

To disable SSH server debugging, use the command:

```
awplus# no debug ssh server
```

Related Commands

debug ssh client
show ssh server
undebug ssh server

service ssh

This command enables the Secure Shell server on the device. Once enabled, connections coming from SSH clients are accepted.

SSH server needs a host key before it starts. If an SSHv2 host key does not exist, then this command fails. If SSHv1 is enabled but a host key for SSHv1 does not exist, then SSH service is unavailable for version 1.

The **no** variant of this command disables the Secure Shell server. When the Secure Shell server is disabled, connections from SSH, SCP, and SFTP clients are not accepted. This command does not affect existing SSH sessions. To terminate existing sessions, use the clear ssh command.

Syntax

```
service ssh [ip|ipv6]

no service ssh [ip|ipv6]
```

Default The Secure Shell server is disabled by default. Both IPv4 and IPv6 Secure Shell server are enabled when you issue **service ssh** without specifying the optional **ip** or **ipv6** parameters.

Mode Global Configuration

Examples To enable both the IPv4 and the IPv6 Secure Shell server, use the commands:

```
        awplus# configure terminal
awplus(config)# service ssh
```

To enable the IPv4 Secure Shell server only, use the commands:

```
        awplus# configure terminal
awplus(config)# service ssh ip
```

To enable the IPv6 Secure Shell server only, use the commands:

```
        awplus# configure terminal
awplus(config)# service ssh ipv6
```

To disable both the IPv4 and the IPv6 Secure Shell server, use the commands:

```
        awplus# configure terminal
awplus(config)# no service ssh
```

To disable the IPv4 Secure Shell server only, use the commands:

```
        awplus# configure terminal
awplus(config)# no service ssh ip
```

To disable the IPv6 Secure Shell server only, use the commands:

```
        awplus# configure terminal
awplus(config)# no service ssh ipv6
```

Related Commands

crypto key generate hostkey
show running-config ssh
show ssh server
ssh server allow-users
ssh server deny-users

show banner login

This command displays the banner message configured on the device. The banner message is displayed to the remote user before user authentication starts.

Syntax

```
show banner login
```

Mode User Exec, Privileged Exec, Global Configuration, Interface Configuration, Line Configuration

Example To display the current login banner message, use the command:

```
awplus# show banner login
```

Related Commands banner login (SSH)

show crypto key hostkey

This command displays the SSH host keys generated by RSA and DSA algorithm.

A host key pair (public and private keys) is needed to enable SSH server. The private key remains on the device secretly. The public key is copied to SSH clients to identify the server

Syntax

```
show crypto key hostkey [dsa|rsa|rsa1]
```

Parameter	Description
dsa	Displays the DSA algorithm public key.
rsa	Displays the RSA algorithm public key for SSH version 2 connections.
rsa1	Displays the RSA algorithm public key for SSH version 1 connections.

Mode User Exec, Privileged Exec and Global Configuration

Output Figure 51-1: Example output from the **show crypto key hostkey** command

```
Type Bits   Fingerprint
-----------------------------------------------------------
rsa  2058   4e:7d:1d:00:75:79:c5:cb:c8:58:2e:f9:29:9c:1f:48
dsa  1024   fa:72:3d:78:35:14:cb:9a:1d:ca:1c:83:2c:7d:08:43
rsa1 1024   e2:1c:c8:8b:d8:6e:19:c8:f4:ec:00:a2:71:4e:85:8b
```

Table 51-1: Parameters in output of the **show crypto key hostkey** command

Parameter	Description
Type	Algorithm used to generate the key.
Bits	Length in bits of the key.
Fingerprint	Checksum value for the public key.

Examples To show the public keys generated on the device for SSH server, use the command:

```
awplus# show crypto key hostkey
```

To display the RSA public key of the SSH server, use the command:

```
awplus# show crypto key hostkey rsa
```

Related Commands crypto key destroy hostkey
crypto key generate hostkey

show crypto key pubkey-chain knownhosts

This command displays the list of public keys maintained in the known host database on the device.

Syntax `show crypto key pubkey-chain knownhosts [<1-65535>]`

Parameter	Description
`<1-65535>`	Key identifier for a specific key. Displays the public key of the entry if specified.

Default Display all keys.

Mode User Exec, Privileged Exec and Global Configuration

Output Figure 51-2: Example output from the **show crypto key public-chain knownhosts** command

```
No     Hostname         Type Fingerprint
-------------------------------------------------------------------------
1      172.16.23.1      rsa  c8:33:b1:fe:6f:d3:8c:81:4e:f7:2a:aa:a5:be:df:18
2      172.16.23.10     rsa  c4:79:86:65:ee:a0:1d:a5:6a:e8:fd:1d:d3:4e:37:bd
3      5ffe:1053:ac21:ff00:0101:bcdf:ffff:0001
                        rsa1 af:4e:b4:a2:26:24:6d:65:20:32:d9:6f:32:06:ba:57
```

Table 51-2: Parameters in the output of the **show crypto key public-chain knownhosts** command

Parameter	Description
`No`	Number ID of the key.
`Hostname`	Host name of the known SSH server.
`Type`	The algorithm used to generate the key.
`Fingerprint`	Checksum value for the public key.

Examples To display public keys of known SSH servers, use the command:

```
awplus# show crypto key pubkey-chain knownhosts
```

To display the key data of the first entry in the known host data, use the command:

```
awplus# show crypto key pubkey-chain knownhosts 1
```

Related Commands crypto key pubkey-chain knownhosts

show crypto key pubkey-chain userkey

This command displays the public keys registered with the SSH server for SSH users. These keys allow remote users to access the device using public key authentication. By using public key authentication, users can access the SSH server without providing password.

Syntax

```
show crypto key pubkey-chain userkey <username> [<1-65535>]
```

Parameter	Description
<username>	User name of the remote SSH user whose keys you wish to display. The username must begin with a letter. Valid characters are all numbers, letters, and the underscore, hyphen and full stop symbols.
<1-65535>	Key identifier for a specific key.

Default Display all keys.

Mode User Exec, Privileged Exec and Global Configuration

Output Figure 51-3: Example output from the **show crypto key public-chain userkey** command

```
No Type Bits Fingerprint
-----------------------------------------------------------------
1  dsa  1024 2b:cc:df:a8:f8:2e:8f:a4:a5:4f:32:ea:67:29:78:fd
2  rsa  2048 6a:ba:22:84:c1:26:42:57:2c:d7:85:c8:06:32:49:0e
```

Table 51-3: Parameters in the output of the **show crypto key userkey** command

Parameter	Description
`No`	Number ID of the key.
`Type`	The algorithm used to generate the key.
`Bits`	Length in bits of the key.
`Fingerprint`	Checksum value for the key.

To display the public keys for the user `manager` that are registered with the SSH server, use the command:

```
awplus# show crypto key pubkey-chain userkey manager
```

Related Commands crypto key pubkey-chain userkey

show crypto key userkey

This command displays the public keys created on this device for the specified SSH user.

Syntax `show crypto key userkey <username> [dsa|rsa|rsa1]`

Parameter	Description
`<username>`	User name of the local SSH user whose keys you wish to display. The username must begin with a letter. Valid characters are all numbers, letters, and the underscore, hyphen and full stop symbols.
`dsa`	Displays the DSA public key.
`rsa`	Displays the RSA public key used for SSH version 2 connections.
`rsa1`	Displays the RSA key used for SSH version 1 connections.

Mode User Exec, Privileged Exec and Global Configuration

Output Figure 51-4: Example output from the **show crypto key userkey** command

```
Type Bits   Fingerprint
-------------------------------------------------------------
rsa  2048   e8:d6:1b:c0:f4:b6:e6:7d:02:2e:a9:d4:a1:ca:3b:11
rsa1 1024   12:25:60:95:64:08:8e:a1:8c:3c:45:1b:44:b9:33:9b
```

Table 51-4: Parameters in the output of the **show crypto key userkey** command

Parameter	Description
`Type`	The algorithm used to generate the key.
`Bits`	Length in bits of the key.
`Fingerprint`	Checksum value for the key.

Examples To show the public key generated for the user, use the command:

```
awplus# show crypto key userkey manager
```

To store the RSA public key generated for the user manager to the file "user.pub", use the command:

```
awplus# show crypto key userkey manager rsa > manager-rsa.pub
```

Related Commands crypto key generate userkey

show running-config ssh

This command displays the current running configuration of Secure Shell (SSH).

Syntax

```
show running-config ssh
```

Mode Privileged Exec and Global Configuration

Output Figure 51-5: Example output from the **show running-config ssh** command

```
!
ssh server session-timeout 600
ssh server login-timeout 30
ssh server allow-users manager 192.168.1.*
ssh server allow-users john
ssh server deny-user john*.a-company.com
ssh server
```

Table 51-5: Parameters in the output of the **show running-config ssh** command

Parameter	Description
`ssh server`	SSH server is enabled.
`ssh server v2`	SSH server is enabled and only support SSHv2.
`ssh server` *`<port>`*	SSH server is enabled and listening on the specified TCP port.
`no ssh server scp`	SCP service is disabled.
`no ssh server sftp`	SFTP service is disabled.
`ssh server session-timeout`	Configure the server session timeout.
`ssh server login-timeout`	Configure the server login timeout.
`ssh server max-startups`	Configure the maximum number of concurrent sessions waiting authentication.
`no ssh server authentication password`	Password authentication is disabled.
`no ssh server authentication publickey`	Public key authentication is disabled.
`ssh server allow-users`	Add the user (and hostname) to the allow list.
`ssh server deny-users`	Add the user (and hostname) to the deny list.

Example To display the current configuration of SSH, use the command:

```
awplus# show running-config ssh
```

Related Commands service ssh

show ssh server

show ssh

This command displays the active SSH sessions on the device, both incoming and outgoing.

Syntax `show ssh`

Mode User Exec, Privileged Exec and Global Configuration

Output Figure 51-6: Example output from the **show ssh** command

```
Secure Shell Sessions:
ID  Type Mode   Peer Host    Username   State       Filename
-----------------------------------------------------------------
414 ssh  server 172.16.23.1  root       open
456 ssh  client 172.16.23.10 manager    user-auth
459 scp  client 172.16.23.12 root       download    550dev_.awd
463 ssh  client 5ffe:33fe:5632:ffbb:bc35:ddee:0101:ac51
                             manager    user-auth
```

Table 51-6: Parameters in the output of the **show ssh** command

Parameter	Description	
`ID`	Unique identifier for each SSH session.	
`Type`	Session type; either SSH, SCP, or SFTP.	
`Mode`	Whether the device is acting as an SSH client (client) or SSH server (server) for the specified session.	
`Peer Host`	The hostname or IP address of the remote server or client.	
`Username`	Login user name of the server.	
`State`	The current state of the SSH session. One of:	
	`connecting`	The device is looking for a remote server.
	`connected`	The device is connected to the remote server.
	`accepted`	The device has accepted a new session.
	`host-auth`	host-to-host authentication is in progress.
	`user-auth`	User authentication is in progress.
	`authenticated`	User authentication is complete.
	`open`	The session is in progress.
	`download`	The user is downloading a file from the device.
	`upload`	The user is uploading a file from the device.
	`closing`	The user is terminating the session.
	`closed`	The session is closed.
`Filename`	Local filename of the file that the user is downloading or uploading.	

Example To display the current SSH sessions on the device, use the command:

```
awplus# show ssh
```

Related Commands clear ssh

show ssh client

This command displays the current configuration of the Secure Shell client.

Syntax `show ssh client`

Mode User Exec, Privileged Exec and Global Configuration

Output Figure 51-7: Example output from the **show ssh client** command

```
Secure Shell Client Configuration
-----------------------------------------------------------------
Port                                : 22
Version                             : 2,1
Connect Timeout                     : 30 seconds
Session Timeout                     : 0 (off)
Debug                               : NONE
```

Table 51-7: Parameters in the output of the **show ssh client** command

Parameter	Description
`Port`	SSH server TCP port where the SSH client connects to. The default is port 22.
`Version`	SSH server version; either "1", "2" or "2,1".
`Connect Timeout`	Time in seconds that the SSH client waits for an SSH session to establish. If the value is 0, the connection is terminated when it reaches the TCP timeout.
`Debug`	Whether debugging is active on the client.

Example To display the current configuration for SSH clients on the login shell, use the command:

```
awplus# show ssh client
```

Related Commands show ssh server

show ssh server

This command displays the current configuration of the Secure Shell server.

Note that changes to the SSH configuration affects only new SSH sessions coming from remote hosts, and does not affect existing sessions.

Syntax `show ssh server`

Mode User Exec, Privileged Exec and Global Configuration

Output Figure 51-8: Example output from the **show ssh server** command

```
Secure Shell Server Configuration
----------------------------------------------------------------
SSH Server                          : Enabled
Port                                : 22
Version                             : 2
Services                            : scp, sftp
User Authentication                 : publickey, password
Idle Timeout                        : 60 seconds
Maximum Startups                    : 10
Debug                               : NONE
```

Table 51-8: Parameters in the output of the **show ssh server** command

Parameter	Description
`SSH Server`	Whether the Secure Shell server is enabled or disabled.
`Port`	TCP port where the Secure Shell server listens for connections. The default is port 22.
`Version`	SSH server version; either "1", "2" or "2,1".
`Services`	List of the available Secure Shell service; one or more of SHELL, SCP or SFTP.
`Authentication`	List of available authentication methods.
`Login Timeout`	Time (in seconds) that the SSH server will wait the SSH session to establish. If the value is 0, the client login will be terminated when TCP timeout reaches.
`Idle Timeout`	Time (in seconds) that the SSH server will wait to receive data from the SSH client. The server disconnects if this timer limit is reached. If set at 0, the idle timer remains off.
`Maximum Startups`	The maximum number of concurrent connections that are waiting authentication. The default is 10.
`Debug`	Whether debugging is active on the server.

Example To display the current configuration of the Secure Shell server, use the command:

```
awplus# show ssh server
```

Related Commands show ssh
show ssh client

show ssh server allow-users

This command displays the user entries in the allow list of the SSH server.

Syntax `show ssh server allow-users`

Mode User Exec, Privileged Exec and Global Configuration

Output Figure 51-9: Example output from the **show ssh server allow-users** command

```
Username          Remote Hostname (pattern)
----------------- --------------------------------
awplus            192.168.*
john
manager           *.alliedtelesis.com
```

Table 51-9: Parameters in the output of the **show ssh server allow-users** command

Parameter	Description
`Username`	User name that is allowed to access the SSH server.
`Remote Hostname (pattern)`	IP address or hostname pattern of the remote client. The user is allowed requests from a host that matches this pattern. If no hostname is specified, the user is allowed from all hosts.

Example To display the user entries in the allow list of the SSH server, use the command:

```
awplus# show ssh server allow-users
```

Related Commands ssh server allow-users
ssh server deny-users

show ssh server deny-users

This command displays the user entries in the deny list of the SSH server. The user in the deny list is rejected to access the SSH server. If a user is not included in the access list of the SSH server, the user is also rejected.

Syntax

```
show ssh server deny-users
```

Mode User Exec, Privileged Exec and Global Configuration

Output Figure 51-10: Example output from the **show ssh server deny-user** command

```
Username          Remote Hostname (pattern)
----------------- --------------------------------
john              *.b-company.com
manager           192.168.2.*
```

Table 51-10: Parameters in the output of the **show ssh server deny-user** command

Parameter	Description
`Username`	The user that this rule applies to.
`Remote Hostname (pattern)`	IP address or hostname pattern of the remote client. The user is denied requests from a host that matches this pattern. If no hostname is specified, the user is denied from all hosts.

Example To display the user entries in the deny list of the SSH server, use the command:

```
awplus# show ssh server deny-users
```

Related Commands ssh server allow-users
ssh server deny-users

ssh

This command initiates a Secure Shell connection to a remote SSH server.

If the server requests a password for the user login, the user needs to type in the correct password on "Password:" prompt.

SSH client identifies the remote SSH server by it's public key registered on the client device. If the server identification is changed, server verification fails. If the public key of the server has been changed, it is required that the public key of the server should be explicitly added to the known host database.

Note Note that any hostname specified with ssh cannot begin with a hyphen (-) character.

Syntax

```
ssh [ip|ipv6][{[user <username>]|[port <1-65535>]|[version {1|2]}}]
    <hostname> [<line>]
```

Parameter	Description	
ip	Specify IPv4 SSH.	
ipv6	Specify IPv6 SSH.	
user	Login user. If user is specified, the username is used for login to the remote SSH server when user authentication is required. Otherwise the current user name is used.	
	<username>	User name to login on the remote server.
port	SSH server port. If port is specified, the SSH client connects to the remote SSH server with the specified TCP port. Other- wise, the client port configured by "ssh client" command or the default TCP port (22) is used.	
	<1-65535>	TCP port.
version	SSH client version. If version is specified, the SSH client supports only the specified SSH version. By default, SSH client uses SSHv2 first. If the server does not support SSHv2, it will try SSHv1. The default version can be configured by "ssh client" command.	
	1	Use SSH version 1.
	2	Use SSH version 2.
<hostname>	IPv4/IPv6 address or hostname of a remote server in the format `a.b.c.d` for an IPv4 address, or in the format `x:x::x:x` for an IPv6 address corresponding to the ip or ipv6 optional keywords used. Note that any hostname specified with ssh cannot begin with a hyphen (-) character.	
	<line>	Command to execute on the remote server. If a command is specified, the command is executed on the remote SSH server and the session is disconnected when the remote command finishes.

Mode User Exec and Privileged Exec

Examples To login to the remote SSH server at 192.0.2.5, use the command:

```
awplus# ssh ip 192.0.2.5
```

To login to the remote SSH server at 192.0.2.5 as user **manager**, use the command:

```
awplus# ssh ip user manager 192.0.2.5
```

To login to the remote SSH server at 192.0.2.5 that is listening TCP port 2000, use the command:

```
awplus# ssh port 2000 192.0.2.5
```

To login to the remote SSH server with example_host using IPv6 session, use the command:

```
awplus# ssh ipv6 example_host
```

To run the **cmd** command on the remote SSH server at 192.0.2.5, use the command:

```
awplus# ssh ip 192.0.2.5 cmd
```

Related Commands crypto key generate userkey
crypto key pubkey-chain knownhosts
debug ssh client
ssh client

ssh client

This command modifies the default configuration parameters of the Secure Shell (SSH) client. The configuration is used for any SSH client on the device to connect to remote SSH servers. Any parameters specified on SSH client explicitly override the default configuration parameters.

The change affects the current user shell only. When the user exits the login session, the configuration does not persist. This command does not affect existing SSH sessions.

The **no** variant of this command resets configuration parameters of the Secure Shell (SSH) client changed by the ssh client command, and restores the defaults.

This command does not affect the existing SSH sessions.

Syntax

```
ssh client {port <1-65535>|version {1|2}|session-timeout <0-3600>|
    connect-timeout <1-600>}

no ssh client {port|version|session-timeout|connect-timeout}
```

Parameter	Description
`port`	The default TCP port of the remote SSH server. If an SSH client specifies an explicit port of the server, it overrides the default TCP port. Default: **22**
	<1-65535> TCP port number.
`version`	The SSH version used by the client for SSH sessions. The SSH client supports both version 2 and version 1 Default: **version 2** **Note**: SSH version 2 is the default SSH version. SSH client supports SSH version 1 if SSH version 2 is not configured using a **ssh version** command.
	`1` SSH clients on the device supports SSH version 1 only.
	`2` SSH clients on the device supports SSH version 2 only
`session-timeout`	The global session timeout for SSH sessions. If the session timer lapses since the last time an SSH client received data from the remote server, the session is terminated. If the value is 0, then the client does not terminate the session. Instead, the connection is terminated when it reaches the TCP timeout. Default: **0** (session timer remains off)
	<0-3600> Timeout in seconds.
`connect-timeout`	The maximum time period that an SSH session can take to become established. The SSH client terminates the SSH session if this timeout expires and the session is still not established. Default: **30**
	<1-600> Timeout in seconds.

Mode Privileged Exec

Examples To configure the default TCP port for SSH clients to 2200, and the session timer to 10 minutes, use the command:

```
awplus# ssh client port 2200 session-timeout 600
```

To configure the connect timeout of SSH client to 10 seconds, use the command:

```
awplus# ssh client connect-timeout 10
```

To restore the connect timeout to its default, use the command:

```
awplus# no ssh client connect-timeout
```

Related Commands show ssh client
ssh

ssh server

This command modifies the configuration of the SSH server. Changing these parameters affects new SSH sessions connecting to the device.

The **no** variant of this command restores the configuration of a specified parameter to its default. The change affects the SSH server immediately if the server is running. Otherwise, the configuration is used when the server starts.

To enable the SSH server, use the service ssh command.

Syntax

```
ssh server {[v1v2|v2only]|<1-65535>}

ssh server {[session-timeout <0-3600>] [login-timeout <1-600>]
    [max-startups <1-128>]}

no ssh server {[session-timeout] [login-timeout] [max-startups]}
```

Parameter	Description	
`v1v2`	Supports both SSHv2 and SSHv1client connections. Default: **v1v2**	
`v2only`	Supports SSHv2 client connections only.	
`<1-65535>`	The TCP port number that the server listens to for incoming SSH sessions. Default: **22**	
`session-timeout`	There is a maximum time period that the server waits before deciding that a session is inactive and should be terminated. The server considers the session inactive when it has not received any data from the client, and when the client does not respond to keep alive messages. Default: **0** (session timer remains off).	
	`<0-3600>`	Timeout in seconds.
`login-timeout`	The maximum time period the server waits before disconnecting an unauthenticated client. Default: **60**	
	`<1-600>`	Timeout in seconds.
`max-startups`	The maximum number of concurrent unauthenticated connections the server accepts. When the number of SSH connections awaiting authentication reaches the limit, the server drops any additional connections until authentication succeeds or the login timer expires for a connection. Default: **10**	
	`<1-128>`	Number of sessions.

Mode Global Configuration

Examples

To configure the session timer of SSH server to 10 minutes (600 seconds), use the commands:

```
        awplus# configure terminal
awplus(config)# ssh server login-timeout 600
```

To configure the login timeout of SSH server to 30 seconds, use the commands:

```
        awplus# configure terminal
awplus(config)# ssh server login-timeout 30
```

To limit the number of SSH client connections waiting authentication from SSH server to 3, use the commands:

```
        awplus# configure terminal
awplus(config)# ssh server max-startups
```

To set max-startups parameters of SSH server to the default configuration, use the commands:

```
        awplus# configure terminal
awplus(config)# no ssh server max-startups
```

To support the Secure Shell server with TCP port 2200, use the commands:

```
        awplus# configure terminal
awplus(config)# ssh server 2200
```

To force the Secure Shell server to support SSHv2 only, use the commands:

```
        awplus# configure terminal
awplus(config)# ssh server v2only
```

To support both SSHv2 and SSHv1, use the commands:

```
        awplus# configure terminal
awplus(config)# ssh server v1v2
```

Related Commands

show ssh server
ssh client

ssh server allow-users

This command adds a username pattern to the allow list of the SSH server. If the user of an incoming SSH session matches the pattern, the session is accepted.

When there are no registered users in the server's database of allowed users, the SSH server does not accept SSH sessions even when enabled.

SSH server also maintains the deny list. The server checks the user in the deny list first. If a user is listed in the deny list, then the user access is denied even if the user is listed in the allow list.

The **no** variant of this command deletes a username pattern from the allow list of the SSH server. To delete an entry from the allow list, the username and hostname pattern should match exactly with the existing entry.

Syntax

```
ssh server allow-users <username-pattern> [<hostname-pattern>]

no ssh server allow-users <username-pattern> [<hostname-pattern>]
```

Parameter	Description
`<username-pattern>`	The username pattern that users can match to. An asterisk acts as a wildcard character that matches any string of characters.
`<hostname-pattern>`	The host name pattern that hosts can match to. If specified, the server allows the user to connect only from hosts matching the pattern. An asterisk acts as a wildcard character that matches any string of characters.

Mode Global Configuration

Example To allow the user `john` to create an SSH session from any host, use the commands:

```
        awplus# configure terminal
awplus(config)# ssh server allow-users john
```

To allow the user `john` to create an SSH session from a range of IP address (from 192.168.1.1 to 192.168.1.255), use the commands:

```
        awplus# configure terminal
awplus(config)# ssh server allow-users john 192.168.1.*
```

To allow the user `john` to create a SSH session from `a-company.com` domain, use the commands:

```
        awplus# configure terminal
awplus(config)# ssh server allow-users john *.a-company.com
```

To delete the existing user entry `john 192.168.1.*` in the allow list, use the commands:

```
        awplus#  configure terminal
 awplus(config)# no ssh server allow-users john 192.168.1.*
```

Related Commands show running-config ssh
show ssh server allow-users
ssh server deny-users

ssh server authentication

This command enables RSA public-key or password user authentication for SSH Server. Apply the **password** keyword with the **ssh server authentication** command to enable password authentication for users. Apply the **publickey** keyword with the **ssh server authentication** command to enable RSA public-key authentication for users.

Use the **no** variant of this command to disable RSA public-key or password user authentication for SSH Server. Apply the **password** keyword with the **no ssh authentication** command to disable password authentication for users. Apply the required **publickey** keyword with the **no ssh authentication** command to disable RSA public-key authentication for users.

Syntax

```
ssh server authentication {password|publickey}
```

```
no ssh server authentication {password|publickey}
```

Parameter	Description
`password`	Specifies user password authentication for SSH server.
`publickey`	Specifies user publickey authentication for SSH server.

Default Both RSA public-key authentication and password authentication are enabled by default.

Mode Global Configuration

Usage For password authentication to authenticate a user, password authentication for a user must be registered in the local user database or on an external RADIUS server, before using the **ssh server authentication password** command.

For RSA public-key authentication to authenticate a user, a public key must be added for the user, before using the **ssh server authentication publickey** command.

Example To enable `password` authentication for users connecting through SSH, use the commands:

```
        awplus# configure terminal
awplus(config)# ssh server authentication password
```

To enable `publickey` authentication for users connecting through SSH, use the commands:

```
        awplus# configure terminal
awplus(config)# ssh server authentication publickey
```

To disable `password` authentication for users connecting through SSH, use the commands:

```
        awplus# configure terminal
awplus(config)# no ssh server authentication password
```

To disable `publickey` authentication for users connecting through SSH, use the commands:

```
        awplus# configure terminal
awplus(config)# no ssh server authentication publickey
```

Related Commands crypto key pubkey-chain userkey
service ssh
show ssh server

ssh server deny-users

This command adds a username pattern to the deny list of the SSH server. If the user of an incoming SSH session matches the pattern, the session is rejected.

SSH server also maintains the allow list. The server checks the user in the deny list first. If a user is listed in the deny list, then the user access is denied even if the user is listed in the allow list.

If a hostname pattern is specified, the user is denied from the hosts matching the pattern.

The **no** variant of this command deletes a username pattern from the deny list of the SSH server. To delete an entry from the deny list, the username and hostname pattern should match exactly with the existing entry.

Syntax

```
ssh server deny-users <username-pattern> [<hostname-pattern>]
```

```
no ssh server deny-users <username-pattern> [<hostname-pattern>]
```

Parameter	Description
<username-pattern>	The username pattern that users can match to. The username must begin with a letter. Valid characters are all numbers, letters, and the underscore, hyphen, full stop and asterisk symbols. An asterisk acts as a wildcard character that matches any string of characters.
<hostname-pattern>	The host name pattern that hosts can match to. If specified, the server denies the user only when they connect from hosts matching the pattern. An asterisk acts as a wildcard character that matches any string of characters.

Mode Global Configuration

Example To deny the user `john` to access SSH login from any host, use the commands:

```
        awplus# configure terminal
awplus(config)# ssh server deny-users john
```

To deny the user john to access SSH login from a range of IP address (from 192.168.2.1 to 192.168.2.255), use the commands:

```
        awplus# configure terminal
awplus(config)# ssh server deny-users john 192.168.2.*
```

To deny the user `john` to access SSH login from `b-company.com` domain, use the commands:

```
        awplus# configure terminal
awplus(config)# ssh server deny-users john*.b-company.com
```

To delete the existing user entry `john 192.168.2.*` in the deny list, use the commands:

```
        awplus# configure terminal
awplus(config)# no ssh server deny-users john 192.168.2.*
```

Related Commands

show running-config ssh
show ssh server deny-users
ssh server allow-users

ssh server resolve-host

This command enables resolving an IP address from a host name using a DNS server for client host authentication.

The **no** variant of this command disables this feature.

Syntax

```
ssh server resolve-hosts
```

```
no ssh server resolve-hosts
```

Default

This feature is disabled by default.

Mode

Global Configuration

Usage

Your device has a DNS Client that is enabled automatically when you add a DNS server to your device. To add a DNS server to the list of servers that the device sends DNS queries to use the ip name-server command on page 25.39.

For information about configuring DNS see "Domain Name System (DNS)" on page 24.8.

Example

To resolve a host name using a DNS server, use the commands:

```
        awplus# configure terminal
awplus(config)# ssh server resolve-hosts
```

Related Commands

ip name-server
show ssh server
ssh server allow-users
ssh server deny-users

ssh server scp

This command enables the Secure Copy (SCP) service on the SSH server. Once enabled, the server accepts SCP requests from remote clients.

You must enable the SSH server as well as this service before the device accepts SCP connections. The SCP service is enabled by default as soon as the SSH server is enabled.

The **no** variant of this command disables the SCP service on the SSH server. Once disabled, SCP requests from remote clients are rejected.

Syntax

```
ssh server scp

no ssh server scp
```

Mode Global Configuration

Examples To enable the SCP service, use the commands:

```
        awplus# configure terminal
awplus(config)# ssh server scp
```

To disable the SCP service, use the commands:

```
        awplus# configure terminal
awplus(config)# no ssh server scp
```

Related Commands show running-config ssh
show ssh server

ssh server sftp

This command enables the Secure FTP (SFTP) service on the SSH server. Once enabled, the server accepts SFTP requests from remote clients.

You must enable the SSH server as well as this service before the device accepts SFTP connections. The SFTP service is enabled by default as soon as the SSH server is enabled. If the SSH server is disabled, SFTP service is unavailable.

The **no** variant of this command disables SFTP service on the SSH server. Once disabled, SFTP requests from remote clients are rejected.

Syntax

```
ssh server sftp

no ssh server sftp
```

Mode Global Configuration

Examples To enable the SFTP service, use the commands:

```
        awplus# configure terminal
awplus(config)# ssh server sftp
```

To disable the SFTP service, use the commands:

```
        awplus# configure terminal
awplus(config)# no ssh server sftp
```

Related Commands show running-config ssh
show ssh server

undebug ssh client

This command applies the functionality of the no debug ssh client command.

undebug ssh server

This command applies the functionality of the no debug ssh server command.

Chapter 52: DHCP Snooping Introduction and Configuration

Introduction

This chapter provides information about DHCP snooping, support for it on this switch, and how to configure it.

For detailed descriptions of the commands used to configure DHCP snooping, see Chapter 53, DHCP Snooping Commands; for related ACL commands, see Chapter 32, IPv4 Hardware Access Control List (ACL) Commands. For information about Dynamic Host Configuration protocol and how to configure it, see Chapter 60, Dynamic Host Configuration Protocol (DHCP) Introduction and Chapter 61, Dynamic Host Configuration Protocol (DHCP) Commands.

DHCP Snooping

DHCP snooping provides an extra layer of security on the switch via dynamic IP source filtering. DHCP snooping filters out traffic received from unknown, or 'untrusted' ports, and builds and maintains a DHCP snooping database.

Dynamic Host Configuration Protocol (DHCP) dynamically assigns IP addresses to client devices. The use of dynamically assigned addresses requires traceability, so that a service provider can determine which clients own a particular IP address at a certain time.

With DHCP snooping, IP sources are dynamically verified, and filtered accordingly. IP packets that are not sourced from recognized IP addresses can be filtered out. This ensures the required traceability.

With DHCP snooping, an administrator can control port-to-IP connectivity by:

- permitting port access to specified IP addresses only
- permitting port access to DHCP issued IP addresses only
- dictating the number of IP clients on any given port
- passing location information about an IP client to the DHCP server
- permitting only known IP clients to ARP

Ports on the switch are classified as either trusted or untrusted:

- Trusted ports receive only messages from within your network.
- Untrusted ports receive messages from outside your network.

DHCP snooping blocks unauthorized IP traffic from untrusted ports, and prevents it from entering the trusted network. It validates DHCP client packets from untrusted ports and forwards them to trusted ports in the VLAN.

On this switch, DHCP snooping is disabled by default, and can be enabled on per-VLAN basis to operate over switch ports and over static and dynamic (LACP) link aggregators (channel groups).

DHCP Snooping Database

When you enable DHCP snooping, the switch intercepts all DHCP packets it receives, and sends them to the Central Processing Unit (CPU), where they are verified. The DHCP snooping database stores and maintains this information. The database contains entries for:

- current IP address leases dynamically allocated by a DHCP server
- static or dynamic entries added from the command line—typically used to add a DHCP snooping entry for a client that has a preconfigured IP address on an untrusted port

Database backup

The switch periodically saves the dynamic entries in the DHCP snooping database to a hidden file (**.dhcp.dsn.gz**) in Non-Volatile Storage (NVS), or can be configured to save it to Flash memory or to a USB storage device. If such a database file exists, it is loaded when the switch starts up with DHCP snooping enabled, or when DHCP snooping is subsequently enabled.

Lease entries

Each entry in the database corresponds to a DHCP IP address lease.

For dynamic entries added automatically by DHCP snooping, each entry contains the following information:

- the IP address that was allocated to that client
- the MAC address of the client device
- the time until expiry
- the VLAN to which the client is attached
- the port to which the client is attached
- the IP address of the DHCP server

For static entries added from the command line, each entry contains the following subset of information:

- the IP address allocated to the client
- the MAC address of the client device (optional)
- the VLAN to which the client is attached
- the port to which the client is attached

Each entry also shows its source: Dynamic or Static.

On this switch, the maximum number of lease entries that can be stored in the DHCP snooping database for each port can be configured—the default is 1.

Expired entries

For dynamic entries, the switch receives expiry information with the client lease information in DHCP packets. Entries expire when the time left to expiry is 0 seconds. Expired entries are automatically deleted from the database. Static entries have no expiry information, and are not checked. All dynamic entries in the database are written to the backup file. Whenever DHCP snooping is enabled, the DHCP snooping database is repopulated from the backup file and any static entries in the start-up configuration file. Any entries present in the backup file that have expired are ignored.

DHCP Option 82

If the switch is at the edge of the network, it can be configured to insert DHCP Option 82 information into client-originated BOOTP/DHCP packets that it is forwarding to a DHCP server. The switch also removes Option 82 information from BOOTP reply packets destined for an untrusted port if the DHCP client hardware is directly attached to a port on the switch.

DHCP servers that are configured to recognize Option 82 may use the information to implement IP address or other parameter assignment policies, based on the network location of the client device.

When Option 82 information for DHCP snooping is enabled, the switch inserts Option 82 information into BOOTP request packets received from an untrusted port. The switch inserts the following Option 82 information:

- Remote ID: this identifies the host. By default, this is the MAC address of the switch (sub-option1).
- Circuit ID: this specifies the switch port and VLAN ID that the client-originated DHCP packet was received on (sub-option2). By default, this is the VLAN ID and the Ifindex (interface number).
- Subscriber ID (optional): this is a string of up to 50 characters that differentiates or groups client ports on the switch (sub-option 6).

Support on this switch

This switch inserts Option 82 (agent option) information into DHCP packets received through untrusted ports, and removes it from DHCP packets transmitted through untrusted ports. This is enabled by default, and can be disabled if required.

You can specify values for the Remote ID and Circuit ID sub-options of the Option 82 field. The Remote ID can be specified as an alphanumeric (ASCII) string, 1 to 63 characters in length. The Circuit ID can be specified as the VLAN ID and port number.

Subscriber IDs can be configured for ports, and if they have been configured, they are inserted in DHCP packets as part of the Option 82 information.

Regardless of whether Option 82 is enabled for DHCP snooping, if the switch receives a BOOTP/DHCP request packet on a trusted port, and the packet contains Option 82 information, it does not update the Option 82 information for the receiver port. By default, if it receives a DHCP request packet containing Option 82 information on an untrusted port, it drops the packet. However, if the switch is connected via untrusted ports to edge switches that insert DHCP Option 82 information into DHCP packets, you may need to allow these DHCP packets through the untrusted ports—the switch can be configured to forward these packets.

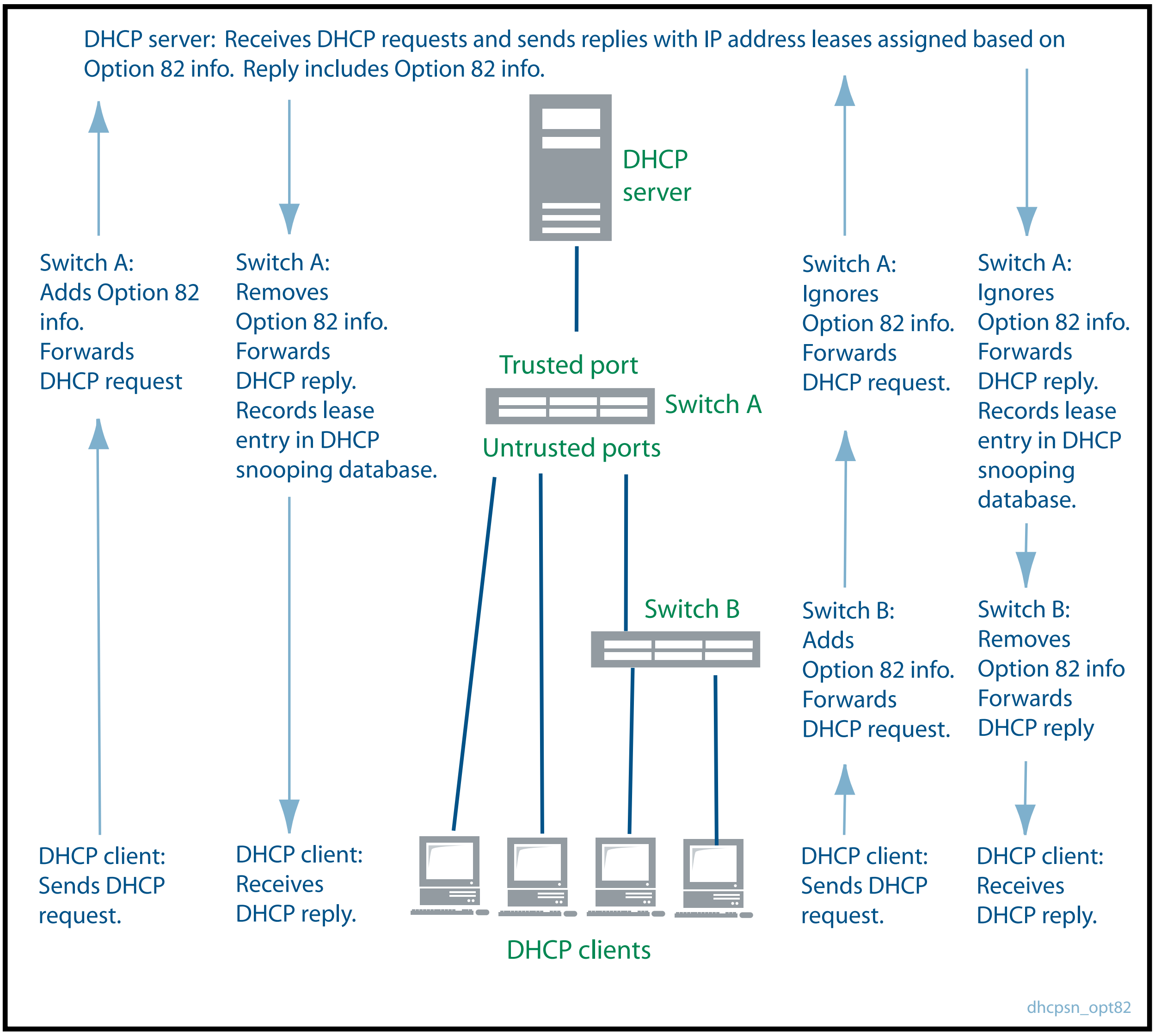
Note that the Option 82 agent information inserted by the DHCP snooping differs from the information added by DHCP Relay (see **“DHCP Relay Agent Introduction” on page 60.8**). The switch cannot be configured to use both the DHCP relay agent option and DHCP snooping.

Operation

Figure 52-1 shows DHCP packet flow between DHCP clients and server, where:

- Switch A has DHCP snooping enabled. The DHCP server is connected to a trusted port on Switch A; DHCP clients and Switch B are connected to untrusted ports.
- Switch A is configured to add and remove DHCP Option 82 information (ip dhcp snooping agent-option command on page 53.12).
- Switch A is configured to forward DHCP packets that already contain Option 82 information without changing it (ip dhcp snooping agent-option allow-untrusted command on page 53.13).
- Switch B is Layer 2 switching traffic from downstream DHCP clients, and adds and removes DHCP Option 82 information.

Figure 52-1: DHCP packet flow with DHCP snooping and Option 82 (agent option)


DHCP server: Receives DHCP requests and sends replies with IP address leases assigned based on Option 82 info. Reply includes Option 82 info.
DHCP server
Switch A: Adds Option 82 info. Forwards DHCP request
Switch A: Removes Option 82 info. Forwards DHCP reply. Records lease entry in DHCP snooping database.
Trusted port
Switch A
Untrusted ports
Switch A: Ignores Option 82 info. Forwards DHCP request.
Switch A: Ignores Option 82 info. Forwards DHCP reply. Records lease entry in DHCP snooping database.
Switch B
Switch B: Adds Option 82 info. Forwards DHCP request.
Switch B: Removes Option 82 info Forwards DHCP reply
DHCP client: Sends DHCP request.
DHCP client: Receives DHCP reply.
DHCP clients
DHCP client: Sends DHCP request.
DHCP client: Receives DHCP reply.
dhcpsn_opt82

For more information about Option 82, see RFC 3046, DHCP Relay Agent Information Option.

Traffic Filtering with DHCP Snooping

DHCP filtering prevents IP addresses from being falsified or 'spoofed'. This guarantees that customers cannot avoid detection by spoofing IP addresses that are not actually allocated to them. With DHCP filtering, the switch permits packets to enter over a specific port if their source IP address is currently allocated to a client connected to that port.

Support on this switch

On this switch, Access Control Lists (ACLs) based on DHCP snooping can be used with access groups to filter IP packets. For instance, IP traffic on untrusted ports can be limited to packets matching valid DHCP lease information stored in the DHCP snooping database. Quality of Service (QoS) configuration can also be applied to these ACLs.

The DHCP snooping feature is enabled or disabled per VLAN, and several of the related configuration settings are applied per port. If there are multiple VLANs on a port, all the VLANs will be subject to the same per-port settings.

Operation

Table 52-1 on page 52.7 shows the filtering that is applied by DHCP snooping on a switch with the following DHCP filtering configuration for untrusted ports:

- DHCP snooping is enabled on all VLANs (service dhcp-snooping command on page 53.26, ip dhcp snooping command on page 53.11)
- ARP security (arp security command on page 53.3) is enabled on all VLANs
- MAC address verification is enabled on the switch (ip dhcp snooping verify mac-address command on page 53.23; enabled by default), and all DHCP clients are directly connected to the switch.
- Access Control Lists allow IP packets that match the source IP address and MAC address of a valid lease entry in the DHCP snooping database, and deny other IP packets (**access-list** commands in Chapter 32, IPv4 Hardware Access Control List (ACL) Commands).
- DHCP requests containing Option 82 info are not allowed (ip dhcp snooping agent-option allow-untrusted command on page 53.13 this is disabled by default).
- Log messages and SNMP notifications are enabled for DHCP snooping and ARP security violations (ip dhcp snooping violation command on page 53.24, arp security violation command on page 53.4, snmp-server enable trap command on page 63.22).

Table 52-1: DHCP filtering on the switch

When the switch ...	And ...	Then the switch ...
DHCP packets		
Receives a DHCP BOOTP packet on a trusted port		Forwards the DHCP packet.
	The packet contains a valid IP address lease for a client, and the maximum number of leases for the client port has not been reached.	Adds or updates a lease entry in the DHCP snooping database.
	The maximum number of leases for the client port has been reached.	Drops the DHCP packet, generates a log message for the violation, generates an SNMP notification (trap), and does not add a lease entry to the database.
A lease entry in the DHCP snooping database expires		Removes the expired entry from the database.
Receives a DHCP BOOTP request packet on an untrusted port	The source MAC address and client hardware address match.	
Receives a DHCP BOOTP request packet on an untrusted port	The source MAC address and client hardware address do not match.	Drops the packet, generates a log message for the violation, and sends an SNMP notification (trap).
Receives a DHCP BOOTP request packet on an untrusted port	The packet contains Option 82 info.	Drops the DHCP packet, generates a log message for the violation, and sends an SNMP notification (trap).
Receives a DHCP BOOTP reply packet on an untrusted port		Drops the DHCP packet, generates a log message for the violation, and sends an SNMP notification (trap).
IP packets		
Receives an IP packet on a trusted port		Forwards the IP packet.
Receives an IP packet on an untrusted port	Its source MAC address, IP address, and receiving port match a valid lease entry in the DHCP snooping database.	Forwards the IP packet.
Receives an IP packet on an untrusted port	Its source MAC address, IP address, and receiving port do not match a valid lease entry in the DHCP snooping database.	Drops the packet. Does not generate a log message or an SNMP notification.
ARP packets		
Receives an ARP request on a trusted port		Forwards the ARP packet.
Receives an ARP request on an untrusted port	Its source MAC address, IP address, and receiving port match a valid entry in the DHCP snooping database	Forwards the ARP packet.
Receives an ARP request on an untrusted port	Its source MAC address, IP address, and receiving port do not match an entry in the DHCP snooping database	Drops the packet, generates a log message for the violation, and sends an SNMP notification (trap).

ARP Security

ARP security prevents ARP spoofing. ARP spoofing occurs when devices send fake, or 'spoofed', ARP messages to an Ethernet LAN. This makes it possible for an unauthorized host to claim to be an authorized host. The unauthorized host can then intercept traffic intended for the authorized host, and can access the wider network.

Spoofed ARP messages contain the IP address of an authorized host, with a MAC address which does not match the real MAC address of the host. When ARP security is enabled for DHCP snooping, the switch checks ARP packets sourced from untrusted ports against the entries in the DHCP snooping binding database. If it finds a matching entry, it forwards the ARP packet as normal. If it does not find a matching entry, it drops the ARP packet. This ensures that only trusted clients (with a recognized IP address and MAC address) can generate ARP packets into the network. ARP security is not applied to packets received on trusted ports.

ARP security is disabled by default, and can be enabled on VLANs to ensure that on untrusted ports, only trusted clients (with a recognized IP address and MAC address) can generate ARP packets into the network. ARP security is applied to both dynamic and static DHCP snooping entries. For static DHCP entries without a MAC address defined, ARP security compares only the IP address details.

MAC Address Verification

When MAC address verification is enabled, the switch forwards DHCP packets received on untrusted ports only if the source MAC address and client hardware address match. MAC address verification is enabled by default.

DHCP Snooping Violations

Packets violating DHCP snooping or ARP security checks (if these are enabled) are automatically dropped. The switch can also be configured to send SNMP notifications (atDhcpsnTrap and atArpsecTrap), to generate log messages, or to shut down the link on which the packet was received.

If the switch is configured to send notifications for DHCP snooping or ARP security violations, the rate is limited to one notification per second. If there are any further violations within a second, no notifications are sent for them. After one second, the switch only sends further notifications if the source MAC address and/or the violation reason are different from previous notifications. (If log messages are also generated for ARP security and DHCP snooping violations, you can see a record of all violations in the log, even if notifications were not sent for all of them.)

Interactions with Other Features

DHCP snooping interacts with other switch features as follows:

- Ports in trunk mode

  The DHCP snooping feature is enabled or disabled per VLAN, and several of the related configuration settings are applied to ports. If there are multiple VLANs on a port, all the VLANs will be subject to the same per-port settings.

- DHCP relay

  The switch cannot use DHCP snooping to filter IP traffic from a DHCP relay device.

  DHCP snooping (**service dhcp-snooping** command on page 53.26) and the DHCP relay agent option (**ip dhcp-relay agent-option** command on page 61.16) cannot both be enabled on the switch at the same time.

- DHCP snooping can be configured with port provisioning.

- Authentication

  DHCP snooping cannot be enabled on a switch that is configured for web authentication (**auth-web enable** command on page 40.27), roaming authentication (**auth roaming enable** command on page 40.16, **auth roaming disconnected** command on page 40.14), or guest VLAN authentication (**auth guest-vlan** command on page 40.8), or vice versa.

- Stacking

  If DHCP snooping is enabled in a stack, the DHCP snooping database and its backup file are automatically synchronized across all stack members, so that a new stack master can reinstate this database.

- Link aggregators

  DHCP snooping can operate over switch ports, and over static and dynamic (LACP) link aggregators (channel groups). If a switch port is added to an aggregator, DHCP snooping configuration is applied to the aggregator; configuration of the original switch port is not preserved. If the switch port is then removed from the aggregator, it returns to default DHCP snooping settings.

- Private VLANs

  Private VLANs are not supported for DHCP snooping.

Configuration

This section provides a general configuration procedure for DHCP snooping.

Configure DHCP Snooping

Note that if a port in trunk mode has multiple VLANs attached, then the DHCP snooping configuration settings for the port apply to all the VLANs.

Table 52-2: General configuration procedure for DHCP snooping

Enable DHCP snooping		
1.	awplus# configure terminal	Enter Global Configuration mode.
2.	awplus(config)# service dhcp-snooping	Enable DHCP snooping on the switch. Default: disabled
3.	awplus(config)# interface <*vid-list*>	Enter Interface Configuration mode for the VLANs to enable DHCP snooping on.
4.	awplus(config-if)# ip dhcp snooping	Enable DHCP snooping on these VLANs. Default: disabled
5.	awplus(config-if)# exit	Return to Global Configuration mode.
6.	awplus(config-if)# interface <*port-list*>	Enter Interface Configuration mode for ports connected to the trusted network. The port(s) connected to the DHCP server(s) must be configured as trusted ports.
7.	awplus(config-if)# ip dhcp snooping trust	Set these ports to be trusted ports. Default: untrusted
8.	awplus(config-if)# exit	Return to Global Configuration mode.
9.	awplus(config)# interface <*port-list*>	If you want to allow more than one DHCP lease for any ports, enter Interface Configuration mode for the required ports. The default is likely to be suitable for edge ports; on an aggregation switch, you may need to increase the maximum number of leases for ports connected to other switches and/or for multiple VLANs. Note that you cannot change this setting once DHCP snooping ACLs are attached to these interfaces.

Table 52-2: General configuration procedure for DHCP snooping(cont.)

10.	**awplus(config-if)#** ip dhcp snooping max-bindings <0-520>	Change the maximum number of leases for these ports. Default: 1
11.	**awplus(config-if)#** exit	Return to Global Configuration mode.
Configure DHCP filtering		
12.	**awplus(config)#** access-list hardware *<name>*	Create a hardware access list, and enter Hardware Access List Configuration mode to configure it. See the access-list hardware (named) command on page 32.17.
13.	**awplus(config-ip-hw-acl)#** [*<seqnum>*] permit ip dhcpsnooping any [*<seqnum>*] deny ip any any	Configure the hardware access list to permit traffic with *source IP address* matching valid entries in the DHCP snooping database, and to deny other traffic. (The last filter applied to the ports by any access list must be the filter that denies all other traffic.) OR
	awplus(config-ip-hw-acl)# [*<seqnum>*] permit ip dhcpsnooping any mac dhcpsnooping any [*<seqnum>*] deny ip any any mac any any	Configure the hardware access list to permit traffic with *source IP address and source MAC address* matching valid entries in the DHCP snooping database, and to deny other traffic. (The last filter applied to the ports by any access list must be the filter that denies all other traffic.) See the (access-list hardware IP protocol filter) command on page 32.22.
14.	**awplus(config-ip-hw-acl)#** exit	Return to Global Configuration mode.
15.	**awplus(config)#** interface *<port-list>*	Enter Interface Configuration mode for the ports to add the DHCP snooping access list to. Typically this would be all untrusted ports.
16.	**awplus(config-if)#** access-group *<name>*	Add the hardware-based access list(s) to these ports. The *name* in this command is the name of the access list specified in Step 12.
17.	**awplus(config-if)#** exit	Return to Global Configuration mode.

Table 52-2: General configuration procedure for DHCP snooping(cont.)

Configure ARP security		
18.	awplus(config)# interface <vid-list>	Enter Interface Configuration mode for the VLANs to enable ARP security on. Default: disabled
19.	awplus(config-if)# arp security	Enable ARP security on particular VLANs if required. On untrusted ports, ARP security forwards ARP packets that have a source IP address and MAC address matching a dynamic entry in the DHCP snooping database, or an IP address matching a static entry. It drops other ARP packets, and treats them as ARP security violations. Default: disabled
20.	awplus(config-if)# exit	Return to Global Configuration mode.
Configure DHCP Option 82		
21.	awplus(config)# no ip dhcp snooping agent-option	If you do not want the switch to insert DHCP Option 82 information into DHCP packets received on untrusted ports, or to remove this information from DHCP packets transmitted on untrusted ports, disable the DHCP Option 82 agent option. Default: enabled if DHCP snooping is enabled.
22.	awplus(config)# ip dhcp snooping agent-option allow-untrusted	If there are edge switches that add the Option 82 information to DHCP packets, and that are connected to untrusted ports on this switch, you may wish to enable this switch to forward these packets, and the associated DHCP reply packets, without changing the Option 82 information in them. Default: disabled.
23.	awplus(config)# interface <port-list>	Enter Interface Configuration mode for one or more ports to add a Subscriber ID for.
24.	awplus(config-if)# ip dhcp snooping subscriber-id [<sub-id>]	Add the Subscriber ID for these ports. The Subscriber ID is included in Option 82 information. Default: no Subscriber ID.
25.	awplus(config)# interface <interface-list>	Enter Interface Configuration mode for one or more VLANs to add a Circuit ID for.

Table 52-2: General configuration procedure for DHCP snooping(cont.)

26.	awplus(config-if)# ip dhcp snooping agent-option circuit-id vlantriplet	Specify the Circuit ID for the VLAN or group of VLANs as the VLAN ID and port number. Default: VLAN ID and Ifindex number.
27.	awplus(config)# interface *<interface-list>*	Enter Interface Configuration mode for one or more VLANs to add a Remote ID for.
28.	awplus(config-if)# ip dhcp snooping agent-option remote-id <remote-id>	Specify the Remote ID for the VLAN or group of VLANs as an alphanumeric (ASCII) string, 1 to 63 characters in length. Default: the switch's MAC address.
29.	awplus(config-if)# exit	Return to Global Configuration mode.
Configure MAC address verification		
30.	awplus(config)# no ip dhcp snooping verify mac-address	If not required, disable MAC address verification. Default: enabled
Configure the DHCP snooping database		
31.	awplus(config)# ip dhcp snooping database {nvs|flash|usb}	If required, change the location of the file to which the switch writes the dynamic entries from the DHCP snooping database. Default: nvs (non-volatile storage)
32.	awplus(config)# no ip dhcp snooping delete-by-client	By default, the switch deletes DHCP lease entries from the DHCP snooping database when it receives matching DHCP release messages. Disable these deletions if required, so that lease entries remain in the database until they expire. Default: enabled—entries are deleted when leases are released.
33.	awplus(config)# ip dhcp snooping delete-by-linkdown	If required, set the switch to delete dynamic entries from the DHCP snooping database when their ports go down. Default: disabled—entries remain if links go down.
34.	awplus(config)# ip source binding <ipaddr> [<macaddr>] vlan <vid> interface <port>	You can actively add, modify, or remove static entries from the DHCP snooping database.
35.	awplus# ip dhcp snooping binding <ipaddr> [<macaddr>] vlan <vid> interface <port> expiry <expiry-time>	You can actively add or remove dynamic entries from the DHCP snooping database. These changes affect the current database and backup file, but are not stored in the running configuration.

Table 52-2: General configuration procedure for DHCP snooping(cont.)

Configure violation actions		
36.	**awplus(config)#** interface *<port-list>*	Enter Interface Configuration mode for the ports for which you want to configure actions in response to DHCP snooping or ARP security violations.
37.	**awplus(config-if)#** ip dhcp snooping violation {log|trap|link-down} ... arp security violation {log|trap|link-down} ...	If required, set the switch to generate an SNMP notification (trap), to generate a log message, and/or to block traffic on the port on which a DHCP snooping and/or ARP security violation is detected. Default: By default, if a packet does not match the DHCP snooping and ARP security restrictions, the packet is dropped, but no other action is taken.
38.	**awplus(config-if)#** exit	Return to Global Configuration mode.
39.	**awplus(config)#** snmp-server enable trap dhcpsnooping	In order to send SNMP notifications: ■ set the action for violations to trap (Step 37) ■ configure SNMP—see Chapter 63, SNMP Commands ■ set the SNMP server to enable DHCP snooping notifications (by default notifications are disabled on the SNMP server). The port connecting the switch to the SNMP manager should be set as a trusted port (Step 7 on page 52.10).
40.	**awplus(config)#** exit	Return to Privileged Exec mode.
Check the configuration		
41.	**awplus#** show ip dhcp snooping show ip dhcp snooping interface [<port-list>] show ip dhcp snooping acl show arp security show arp security interface [<port-list>] show running-config dhcp	Check the DHCP snooping configuration.
Troubleshooting DHCP snooping		
42.	**awplus#** show ip dhcp snooping binding	Check all entries in the DHCP snooping database.
43.	**awplus#** show ip source binding	Check the static entries in the DHCP snooping database.

Table 52-2: General configuration procedure for DHCP snooping(cont.)

44.	awplus#
	`show ip dhcp snooping statistics [detail] [interface <interface-list>]`
	`clear ip dhcp snooping statistics [interface <port-list>]`
		Check DHCP snooping statistics.

45.	awplus#
	`show arp security statistics [detail] [interface <port-list>]`
	`clear arp security statistics [interface <port-list>]`
		Check ARP security statistics.

46.	awplus#
	`debug ip dhcp snooping {all|acl|db|packet [detail]}`
	`show debugging ip dhcp snooping`
	`debug arp security`
	`show debugging arp security`
		Enable debug output for DHCP snooping and/or ARP security.

47.		If you have not already set the switch to log DHCP snooping and ARP security violations, you can do this for troubleshooting purposes. See Step 37 on page 52.14.
48.	awplus# `show log`	Display the contents of the buffered log, including any DHCP snooping log and debug messages. (See also Chapter 10, Logging Commands.)

Disabling DHCP Snooping

If you disable DHCP snooping on the whole switch (no service dhcp-snooping command on page 53.26), all the DHCP snooping configuration is removed, except for the Access Control Lists (ACL). Any ACLs on a port that permit traffic matching DHCP snooping entries and block other traffic, will block all traffic if DHCP snooping is disabled on the port. If you disable DHCP snooping either on the whole switch or on particular VLANs (no ip dhcp snooping command on page 53.11), you must also remove any DHCP snooping ACLs from the ports to maintain connectivity (no access-group command on page 32.4).

Related Features

In addition to configuring DHCP snooping as described in Table 52-2, consider whether you also need to configure the following:

- VLANs—see Chapter 16, VLAN Introduction and Chapter 17, VLAN Commands
- Additional ACL filters—see Chapter 31, Access Control Lists Introduction and Chapter 33, IPv4 Software Access Control List (ACL) Commands
- QoS—see Chapter 35, Quality of Service (QoS) Introduction and Chapter 36, QoS Commands
- SNMP—Chapter 62, SNMP Introduction and Chapter 63, SNMP Commands

Chapter 53: DHCP Snooping Commands

Command List

This chapter gives detailed information about the commands used to configure DHCP snooping. For detailed descriptions of related ACL commands, see Chapter 32, IPv4 Hardware Access Control List (ACL) Commands. For more information about DHCP snooping, see Chapter 52, DHCP Snooping Introduction and Configuration.

DHCP snooping can operate on static link aggregators (e.g., sa2) and dynamic link aggregators (e.g. po3) link aggregators, as well as switch ports (e.g., port1.0.2).

arp security

Use this command to enable ARP security on untrusted ports in the VLANs, so that the switch only responds to/forwards ARP packets if they have recognized IP and MAC source addresses.

Use the **no** variant of this command to disable ARP security on the VLANs.

Syntax

```
arp security

no arp security
```

Default Disabled

Mode Interface Configuration (VLANs)

Usage Enable ARP security to provide protection against ARP spoofing. DHCP snooping must also be enabled on the switch (service dhcp-snooping command on page 53.26), and on the VLANs (ip dhcp snooping command on page 53.11).

Example To enable ARP security on VLANs 2 to 4, use the commands:

```
         awplus# configure terminal
 awplus(config)# interface vlan2-vlan4
awplus(config-if)# arp security
```

Related Commands

arp security violation
show arp security
show arp security interface
show arp security statistics

arp security violation

Use this command to specify an additional action to perform if an ARP security violation is detected on the ports. ARP security must also be enabled (arp security command on page 53.3).

Use the **no** variant of this command to remove the specified action, or all actions. Traffic violating ARP security will be dropped, but no other action will be taken.

Syntax

```
arp security violation {log|trap|link-down} ...
no arp security violation [log|trap|link-down] ...
```

Parameter	Description
`log`	Generate a log message. To display these messages, use the show log command on page 10.37.
`trap`	Generate an SNMP notification (trap). To send SNMP notifications, SNMP must also be configured, and DHCP snooping notifications must be enabled using the snmp-server enable trap command on page 63.22. Notifications are limited to one per second and to one per source MAC and violation reason. Additional violations within a second of a notification being sent will not result in further notifications. Default: disabled.
`link-down`	Shut down the port that received the packet. Default: disabled.

Default When the switch detects an ARP security violation, it drops the packet. By default, it does not perform any other violation actions.

Mode Interface Configuration (switch ports, static or dynamic aggregated links)

Usage When the switch detects an ARP security violation on an untrusted port in a VLAN that has ARP security enabled, it drops the packet. This command sets the switch to perform additional actions in response to ARP violations.

If a port has been shut down in response to a violation, to bring it back up again after any issues have been resolved, use the no shutdown command on page 12.13.

Example To send SNMP notifications for ARP security violations on ports 1.0.1 to 1.0.8, use the commands:

```
          awplus#  configure terminal
  awplus(config)#  snmp-server enable trap dhcpsnooping
  awplus(config)#  interface port1.0.1-port1.0.8
awplus(config-if)# arp security violation trap
```

Related Commands

arp security
show arp security interface
show arp security statistics
show log
snmp-server enable trap

clear arp security statistics

Use this command to clear ARP security statistics for the specified ports, or for all ports.

Syntax

```
clear arp security statistics [interface <port-list>]
```

Parameter	Description
<port-list>	The ports to clear statistics for. If no ports are specified, statistics are cleared for all ports. The ports may be switch ports, or static or dynamic link aggregators.

Mode Privileged Exec

Example To clear statistics for ARP security on interface port1.0.1, use the command:

```
awplus# clear arp security statistics interface port1.0.1
```

Related Commands

arp security violation
show arp security
show arp security statistics

clear ip dhcp snooping binding

Use this command to remove one or more DHCP Snooping dynamic entries from the DHCP Snooping binding database. If no options are specified, all entries are removed from the database.

Caution **If you remove entries from the database for current clients, they will lose IP connectivity until they request and receive a new DHCP lease. If you clear all entries, all clients connected to untrusted ports will lose connectivity.**

Syntax

```
clear ip dhcp snooping binding [<ipaddr>] [interface <port-list>]
    [vlan <vid-list>]
```

Parameter	Description
<ipaddr>	Remove the entry for this client IP address.
<port-list>	Remove all entries for these ports. The port list may contain switch ports, and static or dynamic link aggregators (channel groups).
<vid-list>	Remove all entries associated with these VLANs.

Mode Privileged Exec

Usage This command removes dynamic entries from the database. Note that dynamic entries can also be deleted by using the **no** variant of the ip dhcp snooping binding command on page 53.16.

Dynamic entries can individually restored by using the ip dhcp snooping binding command.

To remove static entries, use the **no** variant of the ip source binding command on page 53.25.

Example To remove a dynamic lease entry from the DHCP snooping database for a client with the IP address 192.168.1.2, use the command:

```
awplus# clear ip dhcp snooping binding 192.168.1.2
```

Related Commands

ip dhcp snooping binding
ip source binding
show ip dhcp snooping binding

clear ip dhcp snooping statistics

Use this command to clear DHCP snooping statistics for the specified ports, or for all ports.

Syntax

```
clear ip dhcp snooping statistics [interface <port-list>]
```

Parameter	Description
<port-list>	The ports to clear statistics for. If no ports are specified, statistics are cleared for all ports. The port list can contain switch ports, or static or dynamic link aggregators.

Mode Privileged Exec

Example To clear statistics for the DHCP snooping on interface port1.0.1, use the command:

```
awplus# clear ip dhcp snooping statistics interface port1.0.1
```

Related Commands

clear arp security statistics
show ip dhcp snooping
show ip dhcp snooping statistics

debug arp security

Use this command to enable ARP security debugging.

Use the **no** variant of this command to disable debugging for ARP security.

Syntax

```
debug arp security
```

```
no debug arp security
```

Default Disabled

Mode Privileged Exec

Example To enable ARP security debugging, use the commands:

```
awplus# debug arp security
```

Related Commands

show debugging arp security
show log
terminal monitor

debug ip dhcp snooping

Use this command to enable the specified types of debugging for DHCP snooping.

Use the **no** variant of this command to disable the specified types of debugging.

Syntax

```
debug ip dhcp snooping {all|acl|db|packet [detail]}
```

```
no debug ip dhcp snooping {all|acl|db|packet [detail]}
```

Parameter	Description
`all`	All DHCP snooping debug.
`acl`	DHCP snooping access list debug.
`db`	DHCP snooping binding database debug.
`packet`	DHCP snooping packet debug. For the **no** variant of this command, this option also disables detailed packet debug, if it was enabled.
`detail`	Detailed packet debug.

Default Disabled

Mode Privileged Exec

Example To enable access list debugging for DHCP snooping, use the commands:

```
awplus# debug ip dhcp snooping acl
```

Related Commands

debug arp security
show debugging ip dhcp snooping
show log
terminal monitor

ip dhcp snooping

Use this command to enable DHCP snooping on one or more VLANs.

Use the **no** variant of this command to disable DHCP snooping on the VLANs.

Syntax

```
ip dhcp snooping

no ip dhcp snooping
```

Default DHCP snooping is disabled on VLANs by default.

Mode Interface Configuration (VLANs)

Usage For DHCP snooping to operate on a VLAN, it must:

- be enabled on the particular VLAN by using this command
- be enabled globally on the switch by using the service dhcp-snooping command on page 53.26
- have at least one port connected to a DHCP server configured as a trusted port by using the ip dhcp snooping trust command on page 53.22

Any ACLs on a port that permit traffic matching DHCP snooping entries and block other traffic, will block all traffic if DHCP snooping is disabled on the port. If you disable DHCP snooping on particular VLANs using this command, you must also remove any DHCP snooping ACLs from the ports to maintain connectivity (no access-group command on page 32.4).

Example To enable DHCP snooping on VLANs 2 to 4, use the commands:

```
           awplus#  configure terminal
   awplus(config)#  interface vlan2-vlan4
awplus(config-if)#  ip dhcp snooping
```

To disable DHCP snooping on the switch, use the command:

```
           awplus#  configure terminal
   awplus(config)#  interface vlan2-vlan4
awplus(config-if)#  no ip dhcp snooping
```

Related Commands

ip dhcp snooping trust
service dhcp-snooping
show ip dhcp snooping

ip dhcp snooping agent-option

Use this command to enable DHCP Option 82 data insertion on the switch. When this is enabled, the switch:

- inserts DHCP Option 82 into DHCP packets that it receives on untrusted ports
- removes DHCP Option 82 from DHCP packets that it sends to untrusted ports.

Use the **no** variant of this command to disable DHCP Option 82.

Syntax

```
ip dhcp snooping agent-option
```

```
no ip dhcp snooping agent-option
```

Default DHCP Option 82 is enabled by default when DHCP snooping is enabled.

Mode Global Configuration

Usage DHCP snooping must also be enabled on the switch (service dhcp-snooping command on page 53.26), and on the VLANs (ip dhcp snooping command on page 53.11).

If a subscriber ID is configured for the port (ip dhcp snooping subscriber-id command on page 53.21), the switch includes this in the DHCP Option 82 information it inserts into DHCP packets received on the port.

Example To disable DHCP Option 82 on the switch, use the commands:

```
        awplus# configure terminal
awplus(config)# no ip dhcp snooping agent-option
```

Related Commands

ip dhcp snooping
ip dhcp snooping agent-option allow-untrusted
ip dhcp snooping subscriber-id
service dhcp-snooping
show ip dhcp snooping

ip dhcp snooping agent-option allow-untrusted

Use this command to enable DHCP Option 82 reception on untrusted ports. When this is enabled, the switch accepts incoming DHCP packets that contain DHCP option 82 data on untrusted ports.

Use the **no** variant of this command to disable DHCP Option 82 reception on untrusted ports.

Syntax

```
ip dhcp snooping agent-option allow-untrusted

no ip dhcp snooping agent-option allow-untrusted
```

Default Disabled

Mode Global Configuration

Usage If the switch is connected via untrusted ports to edge switches that insert DHCP Option 82 data into DHCP packets, you may need to allow these DHCP packets through the untrusted ports, by using this command.

When this is disabled (default), the switch treats incoming DHCP packets on untrusted ports that contain DHCP option 82 data as DHCP snooping violations: it drops them and applies any violation action specified by the ip dhcp snooping violation command on page 53.24. The switch stores statistics for packets dropped; to display these statistics, use the show ip dhcp snooping statistics command on page 53.42.

Example To enable DHCP snooping Option 82 data reception on untrusted ports, use the commands:

```
        awplus#  configure terminal
  awplus(config)#  ip dhcp snooping agent-option allow-untrusted
```

Related Commands

ip dhcp snooping agent-option
ip dhcp snooping violation
show ip dhcp snooping
show ip dhcp snooping statistics

ip dhcp snooping agent-option circuit-id vlantriplet

Use this command to specify the Circuit ID sub-option of the Option 82 field as the VLAN ID and port number. The Circuit ID specifies the switch port and VLAN ID that the client-originated DHCP packet was received on.

Use the **no** variant of this command to set the Circuit ID to the default, the VLAN ID and Ifindex (interface number).

Syntax

```
ip dhcp snooping agent-option circuit-id vlantriplet

no ip dhcp snooping agent-option circuit-id
```

Default By default, the Circuit ID is the VLAN ID and Ifindex (interface number).

Mode Interface Configuration for a VLAN interface.

Usage The Circuit ID sub-option is included in the DHCP Option 82 field of forwarded client DHCP packets:

- DHCP snooping Option 82 is enabled (ip dhcp snooping agent-option command on page 53.12; enabled by default), and
- DHCP snooping is enabled on the switch (service dhcp-snooping) and on the VLAN to which the port belongs (ip dhcp snooping)

Examples To set the Circuit ID to `vlantriplet` for client DHCP packets received on `vlan1`, use the commands:

```
           awplus#  configure terminal
   awplus(config)#  interface vlan1
awplus(config-if)#  ip dhcp snooping agent-option circuit-id
                    vlantriplet
```

To return the Circuit ID format to the default for `vlan1`, use the commands:

```
           awplus#  configure terminal
   awplus(config)#  interface vlan1
awplus(config-if)#  no ip dhcp snooping agent-option circuit-id
```

Related Commands

ip dhcp snooping agent-option
ip dhcp snooping agent-option remote-id
show ip dhcp snooping
show ip dhcp snooping agent-option

ip dhcp snooping agent-option remote-id

Use this command to specify the Remote ID sub-option of the Option 82 field. The Remote ID identifies the device that inserted the Option 82 information. If a Remote ID is not specified, the Remote ID sub-option is set to the switch's MAC address.

Use the **no** variant of this command to set the Remote ID to the default, the switch's MAC address.

Syntax

```
ip dhcp snooping agent-option remote-id <remote-id>

no ip dhcp snooping agent-option remote-id
```

Parameter	Description
<remote-id>	An alphanumeric (ASCII) string, 1 to 63 characters in length. If the Remote ID contains spaces, it must be enclosed in double quotes. Wildcards are not allowed.

Default The Remote ID is set to the switch's MAC address by default.

Mode Interface Configuration for a VLAN interface.

Usage The Remote ID sub-option is included in the DHCP Option 82 field of forwarded client DHCP packets:

- DHCP snooping Option 82 is enabled (ip dhcp snooping agent-option command on page 53.12; enabled by default), and
- DHCP snooping is enabled on the switch (service dhcp-snooping) and on the VLAN to which the port belongs (ip dhcp snooping)

Examples To set the Remote ID to `myid` for client DHCP packets received on `vlan1`, use the commands:

```
          awplus# configure terminal
  awplus(config)# interface vlan1
awplus(config-if)# ip dhcp snooping agent-option remote-id
                   myid
```

To return the Remote ID format to the default for `vlan1`, use the commands:

```
          awplus# configure terminal
  awplus(config)# interface vlan1
awplus(config-if)# no ip dhcp snooping agent-option remote-id
```

Related Commands

ip dhcp snooping agent-option
ip dhcp snooping agent-option circuit-id vlantriplet
show ip dhcp snooping
show ip dhcp snooping agent-option

ip dhcp snooping binding

Use this command to manually add a dynamic-like entry (with an expiry time) to the DHCP snooping database. Once added to the database, this entry is treated as a dynamic entry, and is stored in the DHCP snooping database backup file. This command is not stored in the switch's running configuration.

Use the **no** variant of this command to delete a dynamic entry for an IP address from the DHCP snooping database, or to delete all dynamic entries from the database.

Caution ⚠ **If you remove entries from the database for current clients, they will lose IP connectivity until they request and receive a new DHCP lease. If you clear all entries, all clients connected to untrusted ports will lose connectivity.**

Syntax

```
ip dhcp snooping binding <ipaddr> [<macaddr>] vlan <vid> interface
    <port> expiry <expiry-time>
```

```
no ip dhcp snooping binding [<ipaddr>]
```

Parameter	Description
<ipaddr>	Client's IP address.
<macaddr>	Client's MAC address in HHHH.HHHH.HHHH format.
<vid>	The VLAN ID for the entry, in the range 1 to 4094.
<port>	The port the client is connected to. The port can be a switch port, or a static or dynamic link aggregation (channel group).
<expiry-time>	The expiry time for the entry, in the range 5 to 2147483647 seconds.

Mode Privileged Exec

Usage Note that dynamic entries can also be deleted from the DHCP snooping database by using the clear ip dhcp snooping binding command on page 53.7.

To add or remove static entries from the database, use the ip source binding command on page 53.25.

Example To restore an entry in the DHCP snooping database for a DHCP client with the IP address 192.168.1.2, MAC address 0001.0002.0003, on port1.0.6 of vlan6, and with an expiry time of 1 hour, use the commands:

```
awplus# ip dhcp snooping binding 192.168.1.2 0001.0002.0003
        vlan 6 interface port1.0.6 expiry 3600
```

Related Commands

clear ip dhcp snooping binding
ip source binding
show ip dhcp snooping binding

ip dhcp snooping database

Use this command to set the location of the file to which the dynamic entries in the DHCP snooping database are written. This file provides a backup for the DHCP snooping database.

Use the **no** variant of this command to set the database location back to the default, **nvs**.

Syntax

```
ip dhcp snooping database {nvs|flash|usb}
```

```
no ip dhcp snooping database
```

Parameter	Description
`nvs`	The switch checks the database and writes the file to non-volatile storage (NVS) on the switch at 2 second intervals if it has changed.
`flash`	The switch checks the database and writes the file to Flash memory on the switch at 60 second intervals if it has changed.
`usb`	The switch checks the database and writes the file to a USB storage device installed in the switch at 2 second intervals if it has changed.

Default NVS

Mode Global Configuration

Usage In a stack, the backup file is automatically synchronized across all stack members to the location configured. If the backup file is stored on a USB drive on the stack master, it is only synchronized across stack members that also have a USB drive installed.

If the location of the backup file is changed by using this command, a new file is created in the new location, and the old version of the file remains in the old location. This can be removed if necessary (hidden file: **.dhcp.dsn.gz**).

Example To set the location of the DHCP snooping database to non-volatile storage on the switch, use the commands:

```
        awplus# configure terminal
awplus(config)# ip dhcp snooping database nvs
```

Related Commands show ip dhcp snooping

ip dhcp snooping delete-by-client

Use this command to set the switch to remove a dynamic entry from the DHCP snooping database when it receives a valid DHCP release message with matching IP address, VLAN ID, and client hardware address on an untrusted port, and to discard release messages that do not match an entry in the database.

Use the **no** variant of this command to set the switch to forward DHCP release messages received on untrusted ports without removing any entries from the database.

Syntax

```
ip dhcp snooping delete-by-client

no ip dhcp snooping delete-by-client
```

Default Enabled: by default, DHCP lease entries are deleted from the DHCP snooping database when matching DHCP release messages are received.

Mode Global Configuration

Usage DHCP clients send a release message when they no longer wish to use the IP address they have been allocated by a DHCP server. Use this command to enable DHCP snooping to use the information in these messages to remove entries from its database immediately. Use the **no** variant of this command to ignore these release messages. Lease entries corresponding to ignored DHCP release messages eventually time out when the lease expires.

Example To set the switch to delete DHCP snooping lease entries from the DHCP snooping database when a matching release message is received, use the commands:

```
        awplus# configure terminal
awplus(config)# ip dhcp snooping delete-by-client
```

To set the switch to forward and ignore the content of any DHCP release messages it receives, use the commands:

```
        awplus# configure terminal
awplus(config)# no ip dhcp snooping delete-by-client
```

Related Commands show ip dhcp snooping

ip dhcp snooping delete-by-linkdown

Use this command to set the switch to remove a dynamic entry from the DHCP snooping database when its port goes down. If the port is part of an aggregated link, the entries in the database are only deleted if all the ports in the aggregated link are down.

Use the **no** variant of this command to set the switch not to delete entries when ports go down.

Syntax

```
ip dhcp snooping delete-by-linkdown
```

```
no ip dhcp snooping delete-by-linkdown
```

Default Disabled: by default DHCP Snooping bindings are not deleted when an interface goes down.

Mode Global Configuration

Usage If this command is enabled in a stack, and the master goes down and is replaced by a new master, entries in the DHCP snooping database for ports on the master are removed, unless they are part of link aggregators that are still up.

Examples To set the switch to delete DHCP snooping lease entries from the DHCP snooping database when links go down, use the commands:

```
        awplus# configure terminal
awplus(config)# ip dhcp snooping delete-by-linkdown
```

To set the switch *not* to delete DHCP snooping lease entries from the DHCP snooping database when links go down, use the commands:

```
        awplus# configure terminal
awplus(config)# no ip dhcp snooping delete-by-linkdown
```

Related Commands show ip dhcp snooping

ip dhcp snooping max-bindings

Use this command to set the maximum number of DHCP lease entries that can be stored in the DHCP snooping database for each of the ports. Once this limit has been reached, no further DHCP lease allocations made to devices on the port are stored in the database.

Use the **no** variant of this command to reset the maximum to the default, 1.

Syntax

```
ip dhcp snooping max-bindings <0-520>

no ip dhcp snooping max-bindings
```

Parameter	Description
<0-520>	The maximum number of bindings that will be stored for the port in the DHCP snooping binding database. If 0 is specified, no entries will be stored in the database for the port.

Default The default for maximum bindings is 1.

Mode Interface Configuration (port)

Usage The maximum number of leases cannot be changed for a port while there are DHCP snooping Access Control Lists (ACL) associated with the port. Before using this command, remove any DHCP snooping ACLs associated with the ports. To display ACLs used for DHCP snooping, use the show ip dhcp snooping acl command on page 53.35.

In general, the default (1) will work well on an edge port with a single directly connected DHCP client. If the port is on an aggregation switch that is connected to an edge switch with multiple DHCP clients connected through it, then use this command to increase the number of lease entries for the port.

If there are multiple VLANs configured on the port, the limit is shared between all the VLANs on this port. For example, the default only allows one lease to be stored for one VLAN. To allow connectivity for the other VLANs, use this command to increase the number of lease entries for the port.

Example To set the maximum number of bindings to be stored in the DHCP snooping database to 10 per port for ports 1.0.1 to 1.0.8, use the commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.1-port1.0.8
awplus(config-if)# ip dhcp snooping max-bindings 10
```

Related Commands access-group
show ip dhcp snooping acl
show ip dhcp snooping interface

ip dhcp snooping subscriber-id

Use this command to set a Subscriber ID for the ports.

Use the **no** variant of this command to remove Subscriber IDs from the ports.

Syntax

```
ip dhcp snooping subscriber-id [<sub-id>]

no ip dhcp snooping subscriber-id
```

Parameter	Description
<sub-id>	The Subscriber ID; an alphanumeric (ASCII) string 1 to 50 characters in length. If the Subscriber ID contains spaces, it must be enclosed in double quotes. Wildcards are not allowed.

Default No Subscriber ID.

Mode Interface Configuration (port)

Usage The Subscriber ID sub-option is included in the DHCP Option 82 field of client DHCP packets forwarded from a port if:

- a Subscriber ID is specified for the port using this command, and
- DHCP snooping Option 82 is enabled (ip dhcp snooping agent-option command on page 53.12; enabled by default), and
- DHCP snooping is enabled on the switch (service dhcp-snooping) and on the VLAN to which the port belongs (ip dhcp snooping)

Example To set the Subscriber ID for port 1.0.3 to **room_534**, use the commands:

```
          awplus#  configure terminal
  awplus(config)#  interface port1.0.3
awplus(config-if)# ip dhcp snooping subscriber-id room_534
```

To remove the Subscriber ID from port 1.0.3, use the commands:

```
          awplus#  configure terminal
  awplus(config)#  interface port1.0.3
awplus(config-if)# no ip dhcp snooping subscriber-id
```

Related Commands ip dhcp snooping agent-option
show ip dhcp snooping interface

ip dhcp snooping trust

Use this command to set the ports to be DHCP snooping trusted ports.

Use the **no** variant of this command to return the ports to their default as untrusted ports.

Syntax

```
ip dhcp snooping trust
```

```
no ip dhcp snooping trust
```

Default All ports are untrusted by default.

Mode Interface Configuration (port)

Usage Typically, ports connecting the switch to trusted elements in the network (towards the core) are set as trusted ports, while ports connecting untrusted network elements are set as untrusted. Configure ports connected to DHCP servers as trusted ports.

Example To set switch ports 1.0.1 and 1.0.2 to be trusted ports, use the commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.1-port1.0.2
awplus(config-if)# ip dhcp snooping trust
```

Related Commands show ip dhcp snooping interface

ip dhcp snooping verify mac-address

Use this command to verify that the source MAC address and client hardware address match in DHCP packets received on untrusted ports.

Use the **no** variant of this command to disable MAC address verification.

Syntax

```
ip dhcp snooping verify mac-address

no ip dhcp snooping verify mac-address
```

Default Enabled—source MAC addresses are verified by default.

Mode Global Configuration

Usage When MAC address verification is enabled, the switch treats DHCP packets with source MAC address and client hardware address that do not match as DHCP snooping violations: it drops them and applies any other violation action specified by the ip dhcp snooping violation command on page 53.24. To bring the port back up again after any issues have been resolved, use the no shutdown command on page 12.13.

Example To disable MAC address verification on the switch, use the commands:

```
        awplus# configure terminal
awplus(config)# no ip dhcp snooping verify mac-address
```

Related Commands

ip dhcp snooping violation
show ip dhcp snooping
show ip dhcp snooping statistics

ip dhcp snooping violation

Use this command to specify the action the switch will take when it detects a DHCP snooping violation by a DHCP packet on the ports.

Use the **no** variant of this command to disable the specified violation actions, or all violation actions.

Syntax

```
ip dhcp snooping violation {log|trap|link-down} ...
```

```
no ip dhcp snooping violation [{log|trap|link-down} ...]
```

Parameter	Description
`log`	Generate a log message. To display these messages, use the show log command on page 10.37. Default: disabled.
`trap`	Generate an SNMP notification (trap). To send SNMP notifications, SNMP must also be configured, and DHCP snooping notifications must be enabled using the snmp-server enable trap command on page 63.22. Notifications are limited to one per second and to one per source MAC and violation reason. Default: disabled.
`link-down`	Set the port status to link-down. Default: disabled.

Default By default, DHCP packets that violate DHCP snooping are dropped, but no other violation action is taken.

Mode Interface Configuration (port)

Usage If a port has been shut down in response to a violation, to bring it back up again after any issues have been resolved, use the no shutdown command on page 12.13.

IP packets dropped by DHCP snooping filters do not result in other DHCP snooping violation actions.

Example To set the switch to send an SNMP notification and set the link status to link-down if it detects a DHCP snooping violation on switch ports 1.0.1 to 1.0.4, use the commands:

```
          awplus# configure terminal
  awplus(config)# snmp-server enable trap dhcpsnooping
  awplus(config)# interface port1.0.1-port1.0.4
awplus(config-if)# ip dhcp snooping violation trap link-down
```

Related Commands show ip dhcp snooping interface
show log
snmp-server enable trap

ip source binding

Use this command to add or replace a static entry in the DHCP snooping database.

Use the **no** variant of this command to delete the specified static entry or all static entries from the database.

Syntax

```
ip source binding <ipaddr> [<macaddr>] vlan <vid> interface <port>

no ip source binding [<ipaddr>]
```

Parameter	Description
<ipaddr>	Client's IP address. If there is already an entry in the DHCP snooping database for this IP address, then this command replaces it with the new entry.
<macaddr>	Client's MAC address in HHHH.HHHH.HHHH format.
<vid>	The VLAN ID associated with the entry.
<port>	The port the client is connected to.

Mode Global Configuration

Usage This command removes static entries from the database.

To remove dynamic entries, use the clear ip dhcp snooping binding command on page 53.7 or the **no** variant of the ip dhcp snooping binding command on page 53.16.

Example To add a static entry to the DHCP snooping database for a client with the IP address 192.168.1.2, MAC address 0001.0002.0003, on port1.0.6 of vlan6, use the command:

```
        awplus# configure terminal

awplus(config)# ip source binding 192.168.1.2 0001.0002.0003
                vlan 6 interface port1.0.6
```

To remove the static entry for IP address 192.168.1.2 from the database, use the commands:

```
        awplus# configure terminal

awplus(config)# no ip source binding 192.168.1.2
```

To remove all static entries from the database, use the commands:

```
        awplus# configure terminal

awplus(config)# no ip source binding
```

Related Commands clear ip dhcp snooping binding
ip dhcp snooping binding
show ip dhcp snooping binding
show ip source binding

service dhcp-snooping

Use this command to enable the DHCP snooping service globally on the switch. This must be enabled before other DHCP snooping configuration commands can be entered.

Use the **no** variant of this command to disable the DHCP snooping service on the switch. This removes all DHCP snooping configuration from the running configuration, except for any DHCP snooping maximum bindings settings (ip dhcp snooping max-bindings command on page 53.20), and any DHCP snooping-based Access Control Lists (ACLs), which are retained when the service is disabled.

Syntax

```
service dhcp-snooping

no service dhcp-snooping
```

Default DHCP snooping is disabled on the switch by default.

Mode Global Configuration

Usage For DHCP snooping to operate on a VLAN, it must be enabled on the switch by using this command, and also enabled on the particular VLAN by using the ip dhcp snooping command on page 53.11.

For DHCP snooping to operate on a VLAN, it must:

- be enabled globally on the switch by using this command
- be enabled on the particular VLAN by using the ip dhcp snooping command on page 53.11
- have at least one port connected to a DHCP server configured as a trusted port by using the ip dhcp snooping trust command on page 53.22

If you disable the DHCP snooping service by using the **no** variant of this command, all DHCP snooping configuration (including ARP security, but excluding maximum bindings and ACLs) is removed from the running configuration, and the DHCP snooping database is deleted from active memory. If you re-enable the service, the switch:

- repopulates the DHCP snooping database from the dynamic lease entries in the database backup file (in NVS by default—see the ip dhcp snooping database command on page 53.17). The lease expiry times are updated.

The DHCP snooping service cannot be enabled on a switch that is configured with any of the following features, or vice versa:

- web authentication (auth-web enable command on page 40.27)
- roaming authentication (auth roaming enable command on page 40.16, auth roaming disconnected command on page 40.14)
- guest VLAN authentication (auth guest-vlan command on page 40.8).
- DHCP relay agent option (ip dhcp-relay agent-option command on page 61.16)

Any ACLs on a port that permit traffic matching DHCP snooping entries and block other traffic, will block all traffic if DHCP snooping is disabled on the port. If you disable DHCP snooping on the switch using this command, you must also remove any DHCP snooping ACLs from the ports to maintain connectivity (no access-group command on page 32.4).

Example To enable DHCP snooping on the switch, use the command:

```
        awplus# configure terminal
awplus(config)# service dhcp-snooping
```

To disable DHCP snooping on the switch, use the command:

```
        awplus# configure terminal
awplus(config)# no service dhcp-snooping
```

Related Commands
ip dhcp snooping
ip dhcp snooping database
ip dhcp snooping max-bindings
show ip dhcp snooping

show arp security

Use this command to display ARP security configuration.

Syntax `show arp security`

Mode User Exec and Privileged Exec

Example To display ARP security configuration on the switch use the command:

```
awplus# show arp security
```

Figure 53-1: Example output from the **show arp security** command

```
awplus# show arp security

ARP Security Information:
  Total VLANs enabled ............. 2
  Total VLANs disabled ............ 11
  vlan1 .............. Disabled
  vlan2 .............. Disabled
  vlan3 .............. Disabled
  vlan4 .............. Disabled
  vlan5 .............. Disabled
  vlan100 ............ Disabled
  vlan101 ............ Disabled
  vlan102 ............ Disabled
  vlan103 ............ Disabled
  vlan104 ............ Disabled
  vlan105 ............ Enabled
  vlan1000 ........... Disabled
  vlan1001 ........... Enabled
```

Table 53-1: Parameters in the output from the **show arp security** command

Parameter	Description
`Total VLANs enabled`	The number of VLANs that have ARP security enabled.
`Total VLANs disabled`	The number of VLANs that have ARP security disabled.

Related Commands arp security
show arp security interface
show arp security statistics

show arp security interface

Use this command to display ARP security configuration for the specified ports or all ports.

Syntax `show arp security interface [<port-list>]`

Parameter	Description
<port-list>	The ports to display ARP security information about. The port list can include switch ports, and static or dynamic aggregated links.

Mode User Exec and Privileged Exec

Example To display ARP security configuration for ports, use the command:

```
awplus# show arp security interface
```

Figure 53-2: Example output from the **show arp security interface** command

```
awplus#show arp security interface

Arp Security Port Status and Configuration:

  Port: Provisioned ports marked with brackets, e.g. (portx.y.z)
  KEY:  LG = Log
        TR = Trap
        LD = Link down

Port        Action
-------------------------------
port1.0.1   -- -- --
port1.0.2   -- -- --
port1.0.3   LG TR LD
port1.0.4   LG -- --
port1.0.5   LG -- --
port1.0.6   LG TR --
port1.0.7   LG -- LD
...
```

Table 53-2: Parameters in the output from the **show arp security interface** command

Parameter	Description
`Action`	The action the switch takes when it detects an ARP security violation on the port.
`Port`	The port. Parentheses indicate that ports are configured for provisioning.
`LG, Log`	Generate a log message
`TR, Trap`	Generate an SNMP notification (trap).
`LD, Link down`	Shut down the link.

Related Commands

arp security violation
show arp security
show arp security statistics
show log
snmp-server enable trap

show arp security statistics

Use this command to display ARP security statistics for the specified ports or all ports.

Syntax `show arp security statistics [detail] [interface <port-list>]`

Parameter	Description
`detail`	Display detailed statistics.
`interface <port-list>`	Display statistics for the specified ports.

Mode User Exec and Privileged Exec

Example To display the brief statistics for the ARP security, use the command:

```
awplus# show arp security statistics
```

Figure 53-3: Example output from the **show arp security statistics** command

```
awplus# show arp security statistics

DHCP Snooping ARP Security Statistics:

              In         In
 Interface    Packets    Discards
---------------------------------
 port1.0.3    20         20
 port1.0.4    30         30
 port1.0.12   120        0
```

Table 53-3: Parameters in the output from the **show arp security statistics** command

Parameter	Description
`Interface`	A port name. Parentheses indicate that ports are configured for provisioning.
`In Packets`	The total number of incoming APR packets that are processed by DHCP Snooping ARP Security
`In Discards`	The total number of ARP packets that are dropped by DHCP Snooping ARP Security.

Figure 53-4: Example output from the show arp security statistics detail command

```
awplus#show arp security statistics detail

DHCP Snooping ARP Security Statistics:

Interface ...................... port1.0.3
  In Packets ................... 20
  In Discards .................. 20
    No Lease ................... 20
    Bad Vlan ................... 0
    Bad Port ................... 0
    Source Ip Not Allocated .... 0

Interface ...................... port1.0.4
  In Packets ................... 30
  In Discards .................. 30
    No Lease ................... 30
    Bad Vlan ................... 0
    Bad Port ................... 0
    Source Ip Not Allocated .... 0

Interface ...................... port1.0.12
  In Packets ................... 120
  In Discards .................. 0
    No Lease ................... 0
    Bad Vlan ................... 0
    Bad Port ................... 0
    Source Ip Not Allocated .... 0
```

Related Commands

arp security
arp security violation
clear arp security statistics
show arp security
show arp security interface
show log

show debugging arp security

Use this command to display the ARP security debugging configuration.

Syntax

```
show debugging arp security
```

Mode User and Privileged Exec

Example To display the debugging settings for ARP security on the switch, use the command:

```
awplus# show debugging arp security
```

Figure 53-5: Example output from the **show debugging arp security** command

```
awplus# show debugging arp security

ARP Security debugging status:
  ARP Security debugging is off
```

Related Commands

arp security violation
debug arp security

show debugging ip dhcp snooping

Use this command to display the DHCP snooping debugging configuration.

Syntax `show debugging ip dhcp snooping`

Mode User Exec and Privileged Exec

Example To display the DHCP snooping debugging configuration, use the command:

```
awplus# show debugging ip dhcp snooping
```

Figure 53-6: Example output from the **show debugging ip dhcp snooping** command

```
awplus# show debugging ip dhcp snooping

DHCP snooping debugging status:
  DHCP snooping debugging is off
  DHCP snooping all debugging is off
  DHCP snooping acl debugging is off
  DHCP snooping binding DB debugging is off
  DHCP snooping packet debugging is off
  DHCP snooping detailed packet debugging is off
```

Related Commands debug ip dhcp snooping

show log

show ip dhcp snooping

Use this command to display DHCP snooping global configuration on the switch.

Syntax `show ip dhcp snooping`

Mode User Exec and Privileged Exec

Example To display global DHCP snooping configuration on the switch, use the command:

```
awplus# show ip dhcp snooping
```

Figure 53-7: Example output from the **show ip dhcp snooping** command

```
DHCP Snooping Information:
  DHCP Snooping service ............. Enabled
  Option 82 insertion ............... Enabled
  Option 82 on untrusted ports ...... Not allowed
  Binding delete by client .......... Disabled
  Binding delete by link down ....... Disabled
  Verify MAC address ................ Disabled
  SNMP DHCP Snooping trap ........... Disabled

DHCP Snooping database:
  Database location ................. nvs
  Number of entries in database ..... 2

DHCP Snooping VLANs:
  Total VLANs enabled ............... 1
  Total VLANs disabled .............. 9
  vlan1 .............. Enabled
  vlan2 .............. Disabled
  vlan3 .............. Disabled
  vlan4 .............. Disabled
  vlan5 .............. Disabled
  vlan100 ............ Disabled
  vlan101 ............ Disabled
  vlan105 ............ Disabled
  vlan1000 ........... Disabled
  vlan1001 ........... Disabled
```

Related Commands

service dhcp-snooping
show arp security
show ip dhcp snooping acl
show ip dhcp snooping agent-option
show ip dhcp snooping binding
show ip dhcp snooping interface

show ip dhcp snooping acl

Use this command to display information about the Access Control Lists (ACL) that are using the DHCP snooping database.

Syntax

```
show ip dhcp snooping acl

show ip dhcp snooping acl [detail|hardware] [interface
    [<interface-list>]]
```

Parameter	Description
`detail`	Detailed DHCP Snooping ACL information.
`hardware`	DHCP Snooping hardware ACL information.
`interface`	ACL Interface information.
`<interface-list>`	The interfaces to display information about.

Mode User Exec and Privileged Exec

Example To display DHCP snooping ACL information, use the command:

```
awplus# show ip dhcp snooping acl
```

Figure 53-8: Example output from the **show ip dhcp snooping acl** command

```
awplus#show ip dhcp snooping acl

DHCP Snooping Based Filters Summary:

                           Maximum    Template   Attached
Interface     Bindings   Bindings   Filters    Hardware Filters
-----------------------------------------------------------------
 port1.0.1    1          520        0          0
 port1.0.2    1          3          2          6
 port1.0.3    1          2          4          8
 port1.0.4    1          2          7          14
 port1.0.5    0          2          6          12
 port1.0.6    0          1          0          0
 port1.0.7    0          1          0          0
 port1.0.8    0          1          0          0
 port1.0.9    0          1          0          0
 port1.0.10   0          1          0          0
 port1.0.11   0          1          0          0
 port1.0.12   0          1          0          0
(port2.0.1 )  0          520        0          0
(port2.0.2 )  0          1          0          0
```

To display DHCP snooping hardware ACL information, use the command:

```
awplus# show ip dhcp snooping acl hardware
```

Figure 53-9: Example output from the **show ip dhcp snooping acl hardware** command

```
awplus#show ip dhcp snooping acl detail interface port1.0.4

DHCP Snooping Based Filters in Hardware:

Interface    Access-list(/ClassMap)        Source IP         Source MAC
-----------------------------------------------------------------------------
 port1.0.2   dhcpsn1                       10.10.10.10       aaaa.bbbb.cccc
 port1.0.2   dhcpsn1                       20.20.20.20       0000.aaaa.bbbb
 port1.0.2   dhcpsn1                       0.0.0.0           0000.0000.0000
 port1.0.2   dhcpsn1                       0.0.0.0           0000.0000.0000
 port1.0.2   dhcpsn1                       0.0.0.0           0000.0000.0000
 port1.0.2   dhcpsn1                       0.0.0.0           0000.0000.0000
 port1.0.3   dhcpsn2/cmap1                 30.30.30.30       aaaa.bbbb.dddd
 port1.0.3   dhcpsn2/cmap1                 40.40.40.40       0000.aaaa.cccc
 port1.0.3   dhcpsn2/cmap1                 50.50.50.50       0000.aaaa.dddd
 port1.0.3   dhcpsn2/cmap1                 60.60.60.60       0000.aaaa.eeee
 port1.0.3   dhcpsn2/cmap1                 0.0.0.0           0000.0000.0000
 port1.0.3   dhcpsn2/cmap1                 0.0.0.0           0000.0000.0000
 port1.0.3   dhcpsn2/cmap1                 0.0.0.0           0000.0000.0000
 port1.0.3   dhcpsn2/cmap1                 0.0.0.0           0000.0000.0000
 port1.0.4   dhcpsn3/cmap2                 70.70.70.70
 port1.0.4   dhcpsn3/cmap2                 80.80.80.80
 port1.0.4   dhcpsn2/cmap1                 70.70.70.70
 port1.0.4   dhcpsn2/cmap1                 80.80.80.80
 port1.0.4   dhcpsn1                       70.70.70.70
 port1.0.4   dhcpsn1                       80.80.80.80
```

To display detailed DHCP snooping ACL information for port 1.0.4, use the command:

```
awplus# show ip dhcp snooping acl detail interface port1.0.4
```

Figure 53-10: Example output from the **show ip dhcp snooping acl detail interface** command

```
awplus#show ip dhcp snooping acl detail interface port1.0.4

DHCP Snooping Based Filters Information:

 port1.0.4 : Maximum Bindings ........... 2
 port1.0.4 : Template filters ........... 7
 port1.0.4 : Attached hardware filters .. 14
 port1.0.4 : Current bindings ........... 1, 1 free
 port1.0.4    Client 1 ................ 120.120.120.120
 port1.0.4 : Templates: cheese (via class-map: cmap2)
 port1.0.4 :  10 permit ip dhcpsnooping 100.0.0.0/8
 port1.0.4 : Template: dhcpsn2 (via class-map: cmap1)
 port1.0.4 :  10 permit ip dhcpsnooping any
 port1.0.4 :  20 permit ip dhcpsnooping 10.0.0.0/8
 port1.0.4 :  30 permit ip dhcpsnooping 20.0.0.0/8
 port1.0.4 :  40 permit ip dhcpsnooping 30.0.0.0/8
 port1.0.4 : Template: dhcpsn1 (via access-group)
 port1.0.4 :  10 permit ip dhcpsnooping any mac dhcpsnooping abcd.0000.0000 00
 00.ffff.ffff
 port1.0.4 : 20 permit ip dhcpsnooping any
```

Related Commands access-list hardware (named)
show access-list (IPv4 Hardware ACLs)

show ip dhcp snooping agent-option

Use this command to display DHCP snooping Option 82 information for all interfaces, a specific interface or a range of interfaces.

Syntax

```
show ip dhcp snooping agent-option [interface <interface-list>]
```

Parameter	Description
`interface`	Specify the interface.
`<interface-list>`	The name of the interface or interface range.

Mode User Exec and Privileged Exec

Examples To display DHCP snooping Option 82 information for all interfaces, use the command:

```
awplus# show ip dhcp snooping agent-option
```

To display DHCP snooping Option 82 information for `port1.0.1`, use the command:

```
awplus# show ip dhcp snooping agent-option interface
        port1.0.1
```

To display DHCP snooping Option 82 information for `vlan1`, use the command:

```
awplus# show ip dhcp snooping agent-option interface
        vlan1
```

To display DHCP snooping Option 82 information for `port2.0.1, port4.0.2` and ports in the range from `port4.0.10` to `port4.0.15`, use the command:

```
awplus# show ip dhcp snooping agent-option inteface
        port2.0.1,port4.0.2,port4.0.10-port4.0.15
```

Output Figure 53-11: Example output from the **show ip dhcp snooping agent-option** command

```
awplus#show ip dhcp snooping agent-option

DHCP Snooping Option 82 Configuration:

  Key:     C Id = Circuit Id Format
           R Id = Remote Id
           S Id = Subscriber Id

  Option 82 insertion ............... Enabled
  Option 82 on untrusted ports ...... Not allowed

----------------------------------------------------------------
 vlan1      C Id = vlanifindex
            R Id = Access-Island-01-M1
 vlan2      C Id = vlantriplet
            R Id = Access-Island-01-M1
 vlan3      C Id = vlantriplet
            R Id = Access-Island-01-M3
 vlan4      C Id = vlantriplet
            R Id = 0000.cd28.074c
 vlan5      C Id = vlantriplet
            R Id = 0000.cd28.074c
 vlan6      C Id = vlantriplet
            R Id = 0000.cd28.074c
 port1.0.1  S Id =
 port1.0.2  S Id =
 port1.0.3  S Id = phone_1
 port1.0.4  S Id =
 port1.0.5  S Id =
 port1.0.6  S Id = phone_2
 port1.0.7  S Id = PC_1
 port1.0.8  S Id =
 port1.0.9  S Id =
 port1.0.10 S Id =
 port1.0.11 S Id =
 port1.0.12 S Id =
```

Related Commands

ip dhcp snooping agent-option

ip dhcp snooping agent-option circuit-id vlantriplet

ip dhcp snooping agent-option remote-id

ip dhcp snooping subscriber-id

show ip dhcp snooping

show ip dhcp snooping interface

show ip dhcp snooping binding

Use this command to display all dynamic and static entries in the DHCP snooping binding database.

Syntax `show ip dhcp snooping binding`

Mode User Exec and Privileged Exec

Example To display entries in the DHCP snooping database, use the command:

```
awplus# show ip dhcp snooping binding
```

Figure 53-12: Example output from the **show ip dhcp snooping binding** command

```
awplus# show ip dhcp snooping binding
DHCP Snooping Bindings:

Client           MAC              Server                         Expires
IP Address       Address          IP Address       VLAN  Port    (sec)       Type
--------------------------------------------------------------------------------
1.2.3.4          aaaa.bbbb.cccc   --               7     1.0.10  Infinite    Stat
1.2.3.6          any              --               4077  1.0.10  Infinite    Stat
1.3.4.5          any              --               1     sa1     Infinite    Stat
111.111.100.101  0000.0000.0001   111.112.1.1      1     1.0.10  4076        Dyna
111.111.101.108  0000.0000.0108   111.112.1.1      1     1.0.10  4084        Dyna
111.111.101.109  0000.0000.0109   111.112.1.1      1     1.0.10  4085        Dyna
111.211.100.101  --               --               1     1.0.2   2147483325  Dyna
111.211.100.109  00b0.0000.0009   111.112.111.111  1     1.0.2   21          Dyna
111.211.101.101  00b0.0000.0101   111.112.111.111  1     1.0.2   214         Dyna

Total number of bindings in database: 9
```

Table 53-4: Parameters in the output from the **show ip dhcp snooping binding** command

Parameter	Description
`Client IP Address`	The IP address of the DHCP client.
`MAC Address`	The MAC address of the DHCP client.
`Server IP Address`	The IP address of the DHCP server.
`VLAN`	The VLAN associated with this entry.
`Port`	The port the client is connected to.
`Expires (sec)`	The time in seconds until the lease expires.
`Type`	The source of the entry: ■ Dyna: dynamically entered by snooping DHCP traffic, configured by the ip dhcp snooping binding command, or loaded from the database backup file. ■ Stat: added statically by the ip source binding command
`Total number of bindings in database`	The total number of dynamic and static lease entries in the DHCP snooping database.

Related Commands ip dhcp snooping binding
ip dhcp snooping max-bindings
show ip source binding

show ip dhcp snooping interface

Use this command to display information about DHCP snooping configuration and leases for the specified ports, or all ports.

Syntax

```
show ip dhcp snooping interface [<port-list>]
```

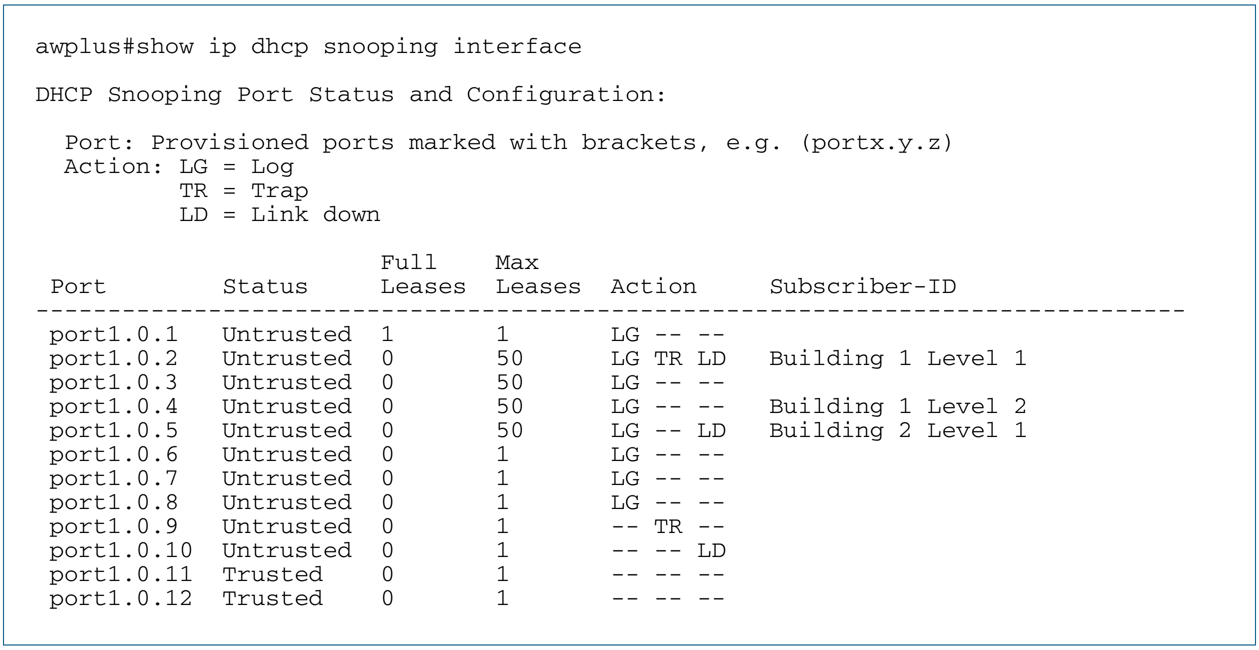
Parameter	Description
<port-list>	The ports to display DHCP snooping configuration information for. If no ports are specified, information for all ports is displayed.

Mode User Exec and Privileged Exec

Example To display DHCP snooping information for all ports, use the command:

```
awplus# show ip dhcp snooping interface
```

Figure 53-13: Example output from the **show ip dhcp snooping interface** command



```
awplus#show ip dhcp snooping interface

DHCP Snooping Port Status and Configuration:

  Port: Provisioned ports marked with brackets, e.g. (portx.y.z)
  Action: LG = Log
          TR = Trap
          LD = Link down

                        Full    Max
 Port        Status     Leases  Leases  Action     Subscriber-ID
--------------------------------------------------------------------------------
 port1.0.1   Untrusted  1       1       LG -- --
 port1.0.2   Untrusted  0       50      LG TR LD   Building 1 Level 1
 port1.0.3   Untrusted  0       50      LG -- --
 port1.0.4   Untrusted  0       50      LG -- --   Building 1 Level 2
 port1.0.5   Untrusted  0       50      LG -- LD   Building 2 Level 1
 port1.0.6   Untrusted  0       1       LG -- --
 port1.0.7   Untrusted  0       1       LG -- --
 port1.0.8   Untrusted  0       1       LG -- --
 port1.0.9   Untrusted  0       1       -- TR --
 port1.0.10  Untrusted  0       1       -- -- LD
 port1.0.11  Trusted    0       1       -- -- --
 port1.0.12  Trusted    0       1       -- -- --
```

Table 53-5: Parameters in the output from the **show ip dhcp snooping interface** command

Parameter	Description
`Port`	The port interface name.
`Status`	The port status: untrusted (default) or trusted.
`Full Leases`	The number of entries in the DHCP snooping database for the port.
`Max Leases`	The maximum number of entries that can be stored in the database for the port.
`Action`	The DHCP snooping violation actions for the port.
`Subscriber ID`	The subscriber ID for the port. If the subscriber ID is longer than 34 characters, only the first 34 characters are displayed. To display the whole subscriber ID, use the show running-config dhcp command on page 7.43.

Related Commands

show ip dhcp snooping
show ip dhcp snooping statistics
show running-config dhcp

show ip dhcp snooping statistics

Use this command to display DHCP snooping statistics.

Syntax

```
show ip dhcp snooping statistics [detail] [interface
    <interface-list>]
```

Parameter	Description
`detail`	Display detailed statistics.
`interface <interface-list>`	Display statistics for the specified interfaces. The interface list can contain switch ports, static or dynamic link aggregators (channel groups), or VLANs.

Mode User Exec and Privileged Exec

Example To show the current DHCP snooping statistics for all interfaces, use the command:

```
awplus# show ip dhcp snooping statistics
```

Figure 53-14: Example output from the **show ip dhcp snooping statistics** command

```
awplus# show ip dhcp snooping statistics

DHCP Snooping Statistics:

               In            In BOOTP      In BOOTP      In
 Interface     Packets       Requests      Replies       Discards
---------------------------------------------------------------
 vlan1         444           386           58            223
 port1.0.1     386           386           0             223
 port1.0.2     0             0             0             0
 port1.0.3     0             0             0             0
 port1.0.4     0             0             0             0
 port1.0.5     0             0             0             0
 port1.0.6     0             0             0             0
 port1.0.7     0             0             0             0
 port1.0.8     0             0             0             0
 port1.0.9     0             0             0             0
 port1.0.10    0             0             0             0
 port1.0.11    0             0             0             0
 port1.0.12    58            0             58            0
```

Figure 53-15: Example output from the **show ip dhcp snooping statistics detail** command

```
awplus# show ip dhcp snooping statistics detail

DHCP Snooping Statistics:

Interface ....................................... port1.0.1, All counters 0
Interface ....................................... port1.0.2, All counters 0
Interface ....................................... port1.0.3, All counters 0
Interface ....................................... port1.0.4
  In Packets .................................... 50
    In BOOTP Requests ........................... 25
    In BOOTP Replies ............................ 25
  In Discards ................................... 1
    Invalid BOOTP Information ................... 0
    Invalid DHCP ACK ............................ 0
    Invalid DHCP Release or Decline ............. 0
    Invalid IP/UDP Header ....................... 0
    Max Bindings Exceeded ....................... 1
    Option 82 Insert Error ...................... 0
    Option 82 Received Invalid .................. 0
    Option 82 Received On Untrusted Port ........ 0
    Option 82 Transmit On Untrusted Port ........ 0
    Reply Received On Untrusted Port ............ 0
    Source MAC/CHADDR Mismatch .................. 0
    Static Entry Already Exists ................. 0
Interface ....................................... port1.0.5, All counters 0
Interface ....................................... port1.0.6, All counters 0
Interface ....................................... port1.0.7, All counters 0
Interface ....................................... port1.0.8, All counters 0
Interface ....................................... port1.0.9, All counters 0
Interface ....................................... port1.0.10, All counters 0
Interface ....................................... port1.0.11, All counters 0
Interface ....................................... port1.0.12, All counters 0
```

Table 53-6: Parameters in the output from the **show ip dhcp snooping statistics** command

Parameter	Description
`Interface`	The interface name.
`In Packets`	The total number of incoming packets that are processed by DHCP Snooping.
`In BOOTP Requests`	The total number of incoming BOOTP Requests.
`In BOOTP Replies`	The total number of incoming BOOTP Replies.
`In Discards`	The total number of incoming packets that have been discarded.
`Invalid BOOTP Information`	Packet contained invalid BOOTP information, such as an invalid BOOTP.OPCode.
`Invalid DHCP ACK`	A DHCP ACK message was discarded, for reasons such as missing Server Option or Lease Option.
`Invalid DHCP Release or Decline`	A DHCP Release or Decline message was discarded, for reasons such as mismatch between received interface and current binding information.
`Invalid IP/UDP Header`	A problem was detected in the IP or UDP header of the packet.
`Max Bindings Exceeded`	Accepting the packet would cause the maximum number of bindings on a port to be exceeded.
`Option 82 Insert Error`	An error occurred while trying to insert option 82 information.

Table 53-6: Parameters in the output from the **show ip dhcp snooping statistics** command(cont.)

Parameter	Description
`Option 82 Received Invalid`	The option 82 information received did not match the information inserted by DHCP Snooping.
`Option 82 Received On Untrusted Port`	A packet containing option 82 was received on an untrusted port.
`Option 82 Transmit On Untrusted Port`	A packet containing option 82 was to be sent on an untrusted port.
`Reply Received On Untrusted Port`	A BOOTP reply was received on an untrusted port.
`Source MAC/CHADDR Mismatch`	The L2 Source MAC address of the packet did not match the client hardware address field (BOOTP.CHADDR).
`Static Entry Already Exists`	An entry could not be added as a static entry already exists.

Related Commands

clear ip dhcp snooping statistics
ip dhcp snooping
ip dhcp snooping violation

show ip source binding

Use this command to display static entries in the DHCP snooping database. These are the entries that have been added by using the ip source binding command on page 53.25.

Syntax

```
show ip source binding
```

Mode

User Exec and Privileged Exec

Example

To display static entries in the DHCP snooping database information, use the command:

```
awplus# show ip source binding
```

Figure 53-16: Example output from the **show ip source binding** command

```
awplus# show ip source binding

IP Source Bindings:

Client        MAC                                 Expires
IP Address    Address         VLAN  Port          (sec)      Type
----------------------------------------------------------------------
 1.1.1.1      0000.1111.2222  1     port1.0.1     Infinite   Static
```

Table 53-7: Parameters in the output from the **show ip source binding** command

Parameter	Description
`Client IP Address`	The IP address of the DHCP client.
`MAC Address`	The MAC address of the DHCP client.
`VLAN`	The VLAN ID the packet is received on.
`Port`	The Layer 2 port name the packet is received on.
`Expires (sec)`	Always infinite for static bindings, or when the leave time in the DHCP message was 0xffffffff (infinite).
`Type`	DHCP Snooping binding type: Static

Related Commands

ip source binding
show ip dhcp snooping binding

Part 6: Network Availability

Chapter 54: VRRP Introduction and Configuration

Introduction

This chapter describes the Virtual Router Redundancy Protocol (VRRP) feature provided by the switch, and how to configure the switch to participate in a virtual router.

The Virtual Router Redundancy Protocol defined in RFC 2338 provides a solution to the problem by combining two or more physical switches into a logical grouping called a **virtual router** (VR). The physical switches then operate together to provide a single logical gateway for hosts on the LAN.

Note If there are PIM-SM routers using VRRP the Bootstrap Router (BSR) function will not work properly.

Virtual Router Redundancy Protocol

The virtual router has a virtual MAC address that is known by all its participating switches or routers. The virtual MAC address is derived from the virtual router identifier - a user-defined value from 1 to 255. At the network level, all hosts on the LAN are configured with a common IP address that is used as the first hop. This IP address is typically owned by the virtual router's preferred individual switch or router. When available, this device performs the duties of the virtual router, and is referred to as the **master**. The switch that owns the IP address associated with the virtual router is referred to as the **preferred master**. When a virtual router is configured so that none of the participating switches owns the IP address, the virtual router has no preferred master.

When a switch takes the role of master for a virtual router, it is responsible for the following:

- Responding to ARP packets that contain IP addresses associated with the virtual router. The ARP response contains the virtual MAC address of the virtual router so that the hosts on the LAN associate the virtual MAC address with their configured first-hop IP address.
- Forwarding packets with a destination link layer MAC address equal to the virtual router MAC address.
- Accepting packets addressed to the IP addresses associated with the virtual router, but only if it actually owns the address(es).
- Broadcasting advertisement packets at regular intervals (at the specified advertisement interval) to inform backup switches that it is still acting as the master switch.

In accordance with the RFC standard, a user does not receive a response to ping or Telnet packets sent to the VR address unless the switch owns this address.

Each of the other switches participating in the virtual router is considered to be a backup switch. A switch can be part of several different virtual routers on one LAN, but all the virtual routers must have different virtual router identifiers (VRID). When a switch has the role of backup for a virtual router, it must be able to perform the following tasks:

- Receive advertisement packets from the master and check that the information contained in them is consistent with their own configuration; ignoring and discarding advertisement packets that do not match.
- Assume the role of master for the virtual router if an advertisement packet is not received for a given period, (the master-down time), based on the specified advertisement interval, (for example: **awplus(config-router)# advertisement-interval 5** will set the advertisement-interval to 5 seconds). The master-down time is approximately three times the advertisement interval.
- Assume the role of master if it receives an advertisement packet from another switch with a lower priority than its own, and if preempt mode is on.

If the master switch fails, the backup switch assumes control and starts processing traffic.
If a backup switch is about to assume the role of master of the VR because it has not received an advertisement for the master-down period, it first checks the operational status of the interface to which the VR is attached. If the interface is down, it does not enter the master state. Instead, it stays in the backup state and checks the interface again after another master-down period, assuming it does not receive an advertisement during that time.

VRRP Configuration

VRRP is disabled by default. Once you have defined a virtual router session, you must enable VRRP to make the session operational for a given interface. You can then enable or disable the virtual router as shown:

To enable VRRP

awplus(config)# router vrrp 1 vlan2	Create a new VRRP session on the router, specify the virtual router ID (VRID) for the session, and specify the interface (vlan2) that will participate in virtual routing.
awplus(config-router)# enable	Enable the VRRP session on the switch.
awplus(config-router)# exit	Return to the Global Configuration mode.
awplus(config)#	Global Configuration mode prompt.

To disable VRRP

awplus(config)# router vrrp 1 vlan2	Specify an existing VRRP session, specify the virtual router ID (VRID) for the session, and specify the interface (vlan2) that will participate in virtual routing.
awplus(config-router)# disable	Disable the VRRP session on the switch.
awplus(config-router)# exit	Return to the Global Configuration mode.
awplus(config)#	Global Configuration mode prompt.

A virtual router must be defined on at least two switches before it operates correctly. Use the following steps to configure virtual routing on a switch. Note that this example assumes that VLAN 2 already exists on the switch. See "Configuring VLANs" on page 16.3.

To configure virtual routing on a switch

awplus# configure terminal	Enter the Global Configuration mode.
awplus(config)# router vrrp 1 vlan2	Create a new VRRP session on the router, specify the VRID for the session, and specify the interface (vlan2) that will participate in virtual routing.

To configure virtual routing on a switch(cont.)

awplus(config-router)# virtual-ip 10.10.10.50 master	Set the virtual IP address for the VRRP session. Define the default state (master or backup) of the VRRP router within the virtual router.
awplus(config-router)# priority 255	Set the VRRP priority for the switch.
awplus(config-router)# enable	Enable the VRRP session on the switch.
awplus(config-router)# exit	Return to the Global Configuration mode.
awplus(config)#	Global Configuration mode prompt.

To destroy a virtual router on the LAN, it must be removed from all participating switches. Use the following commands to remove a virtual router so that the switch no longer participates in virtual routing.

To remove the virtual router VRRP 1 from a switch

awplus# configure terminal	Enter the Global Configuration mode.
awplus(config)# no router vrrp 1 vlan2	Remove the VRRP session on the switch for the specified interface vlan2.
awplus(config-router)# exit	Return to the Global Configuration mode.
awplus(config)#	Global Configuration mode prompt.

Alternatively, you can simply disable the virtual router and retain the configuration.

To disable the router and retain the configuration

awplus# configure terminal	Enter the Global Configuration mode.
awplus(config)# router vrrp 1 vlan2	Select the VRRP session on the switch, specify the VRID for the session, and specify the interface (vlan2) used for virtual routing.
awplus(config-router)# disable	Disable the VRRP session on the switch.
awplus(config-router)# exit	Return to the Global Configuration mode.
awplus(config)#	Global Configuration prompt.

VRRP election and preempt

If the switch that is the current VRRP master becomes unavailable, the master role is taken by the switch with the next highest priority. The priority is a value from 1 to 255, with a default of 100. The value 255 is reserved for the switch that owns the virtual router's IP address. The new master takes over all the responsibilities of the original master.

By default, when a switch becomes available that has a higher priority than the master, this switch takes over as master. This is referred to as **preempt mode** and can be set **on** or **off**. Even with preempt mode **off**, the switch that owns the IP address always becomes the master when available. Preempt mode should be the same for all switches in the virtual router.

If two switches are configured with the same priority and a conflict occurs when they both transition to master simultaneously, the one with the highest IP address has higher priority. Due to timing differences the conflict may not always occur and simply the first switch to respond will become the master.

Hosts on the LAN can continue sending packets to the virtual MAC address they originally associated with the first hop IP address, even though the switch that owns the IP address is not currently available. When the original switch becomes available again, and if it is a preferred switch (i.e. it owns the virtual router IP address) then it resumes the role of master.

Use the following commands to set the priority and preempt mode when you create the virtual router:

To set the priority and preempt mode for VRRP 1

`awplus#`
`configure terminal`
	Enter the Global Configuration mode.

`awplus(config)#`
`router vrrp 1 vlan2`
	Select the VRRP session on the switch, specify the VRID for the session, and specify the interface (`vlan2`) used for virtual routing.

`awplus(config-router)#`
`priority 255`
	Set the VRRP priority for the switch

`awplus(config-router)#`
`preempt true`
	Select the preempt mode for VRRP 1.

`awplus(config-router)#`
`enable`
	Enable the VRRP session on the switch.

`awplus(config-router)#`
`exit`
	Return to the Global Configuration mode.

`awplus(config)#`	Global Configuration prompt

The advertisement interval determines the rate that the master sends its advertisement packets. This rate must be the same value for all switches in the virtual router. The default advertisement interval of 1second can be used for most networks. However, you can modify this interval by using the advertisement-interval command, as shown in the following procedure:

To set the advertisement interval to 5 seconds on VRRP1

awplus# configure terminal	Enter the Global Configuration mode.
awplus(config# router vrrp 1 vlan2	Select the VRRP session on the switch, specify the VRID for the session, and specify the interface (vlan2) used for virtual routing.
awplus(config-router)# advertisement-interval 5	Set the advertisement interval to 5 seconds.

VRRP authentication

Each of the switches in the virtual router can be configured for plaintext authentication, or no authentication. Authentication is appropriate where there is either a security risk, or the configuration is complex.

Plaintext password authentication protects against accidental miscommunication and prevents a switch from inadvertently backing up another switch. This kind of miscommunication could occur, for example, where multiple virtual routers exist on the same LAN.

The authentication type and, in the case of plaintext authentication, the password, must be the same for all switches in the virtual router. By default, the virtual router has no authentication. Authentication must be defined against the relevant interface in the interface configuration mode as shown: This example assumes that VLAN 2 already exists on the switch. See "Configuring VLANs" on page 16.3.

To set the authentication string "guest" to VLAN 2

awplus(config)# router vrrp 1 vlan2	Enable VRRP on the switch, specify the VRID for the session, and specify the interface (vlan2) used for virtual routing.
awplus(config-if)# ip vrrp authentication mode text	Apply text mode authentication to interface vlan2.
awplus(config-if)# ip vrrp authentication string guest	Specify the authentication string or password used by the key.
Switch_A(config-if)# exit	Exit the Interface Configuration mode and enter the Global Configuration mode.
awplus(config)# exit	Return to the Privileged Exec mode prompt
awplus#	Privileged Exec mode prompt

In order to maintain consistent authentication level, each switch in the virtual router must have at least the minimum allowable level of security that meets the network environment.

VRRP debugging

VRRP debugging displays data that is useful for troubleshooting. To enable or disable debugging use the following commands:

To select and deselect VRRP debugging

awplus# `configure terminal`	Enter the Global Configuration mode.
awplus(config)# `debug vrrp [all|events|packet]`	Enable the selected debugging type.
awplus(config)# `no debug vrrp [all|events|packet]`	Disable the selected debugging type.

It is important that all switches involved in a virtual router are configured with the same values for the following:

- VRRP virtual router identifier
- IP address
- advertisement interval
- preempt mode
- authentication type
- password

Inconsistent configuration causes advertisement packets to be rejected and the virtual router cannot perform properly.

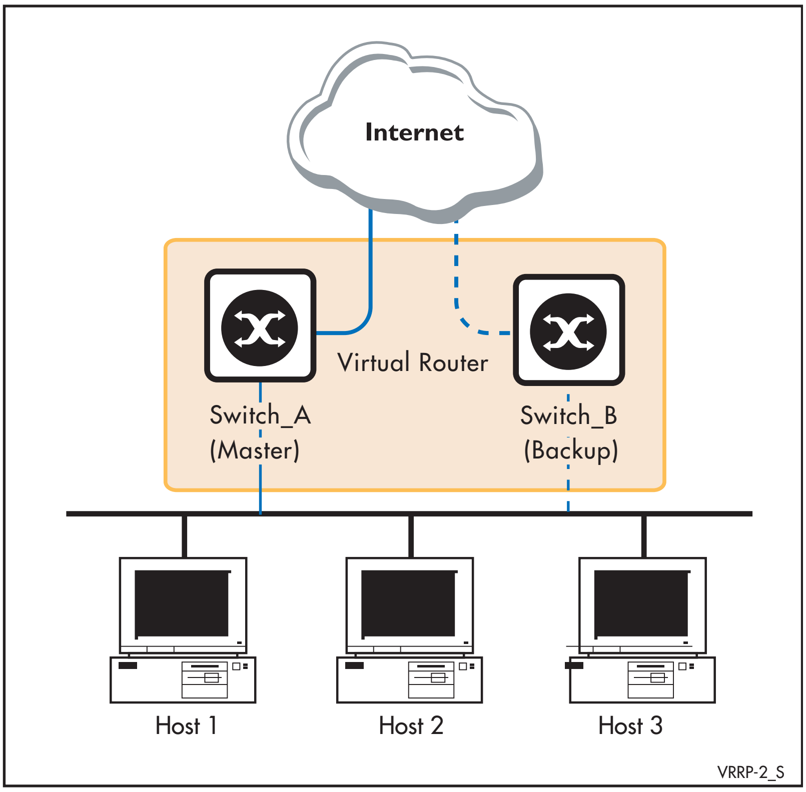
Configuration examples

The following examples show how to configure a virtual router in a LAN:

- Preferred Master with Backup Switch
- Authenticated Virtual Router with No Preferred Master

Master with backup switch

This example show how to configure a basic virtual router with a preferred master and a backup.



Switch_A owns the IP address of the virtual router, and always assumes the role of master whenever it is available. Switch_B is the backup, and assumes the role of master, backing up this IP address if A becomes unavailable. No authentication is used for this simple virtual router.

Step 1: Configure Switch_A

At this point we assume that you have already created VLAN 2 on Switch_A. See “Configuring VLANs” on page 16.3.

Configure IP

awplus# `configure terminal`	Enter the Global Configuration mode.
awplus(config)# `hostname Switch_A`	Assign a host name to `Switch_A`.
Switch_A(config)# `interface vlan2`	Specify the interface (`vlan2`) that will participate in virtual routing to first configure an IP address for `vlan2`.
Switch_A(config-if)# `ip address 192.168.1.1/24`	Specify the IP address and mask for interface `vlan2`.

Create the Virtual Router

Switch_A(config)# `spanning-tree mode stp`	Configure STP for interfaces on `Switch_A`.
Switch_A(config)# `router vrrp 1 vlan2`	Create a new VRRP session on the router, specify the VRID for the session, and specify the interface (`vlan2`) that will participate in virtual routing.
Switch_A(config-router)# `virtual-ip 192.168.1.1 master`	Set the virtual IP address for the VRRP session. Define the default state of the VRRP router within the virtual router.
Switch_A(config-router)# `enable`	Enable the VRRP session on the router.
Switch_A(config-router)# `exit`	Exit the Router Configuration mode and enter the Global Configuration mode.
Switch_A(config)# `exit`	Exit the Global Configuration mode and enter the Privileged Exec mode.
Switch_A#	Privileged Exec mode prompt.

Step 2: Configure Switch_B

At this point we assume that you have already created VLAN 2 on Switch_B. See "Configuring VLANs" on page 16.3.

Configure IP

awplus# configure terminal	Enter Global Configuration mode.
awplus(config)# hostname Switch_B	Assign a host name to Switch_B.
Switch_B(config)# interface vlan2	Specify the interface (vlan2) that will participate in virtual routing.
Switch_B(config)# ip address 192.168.1.2/24	Specify the IP address and mask for interface

Create the Virtual Router

Switch_B(config)# router vrrp 1 vlan2	Create a new VRRP session on the router, specify the VRID for the session, and specify the interface (vlan2) that will participate in virtual routing.
Switch_B(config-router)# virtual-ip 192.168.1.1 backup	Set the virtual IP address for the VRRP session. Define the default state of the VRRP router within the virtual router.
Switch_B(config-router)# enable	Enable the VRRP session on the router.
Switch_B(config-router) exit	Exit the Interface Configuration mode and enter the Global Configuration mode.
Switch_B(config)# exit	Return to the Privileged Exec mode.
Switch_B#	Privileged Exec mode prompt.

Authenticated Virtual Router with an Independent Preferred Master

This example shows how to configure a virtual router with its own IP address. The address is not owned by any of the switches participating in the virtual router. Switch A has a higher priority for becoming the master, Switch B has the next highest priority, and Switch C takes the master role when A or B are unavailable. The default preempt mode (preempt on) ensures that the switch with the highest priority (when it is available) always takes the master role from a lower priority switch acting as master. Plaintext authentication protects against accidental misconfiguration.

Although the switch with the highest priority will be master, it is important to remember that when creating VRRP you must also define the default role. In the following example Switch_A will be defined as being the master.

At this point we assume that you have already created VLAN 2 on Switches A, B and C. See "Configuring VLANs" on page 16.3.

Step 1: Configure IP

On switch_A, add an IP interface to the virtual router.

`awplus#`
`configure terminal`
	Enter Global Configuration mode.

`awplus(config)#`
`hostname Switch_A`
	Assign a host name to `Switch_A`.

`Switch_A(config)#`
`interface vlan2`
	Specify the interface (`vlan2`) that will participate in virtual routing.

`Switch_A(config)#`
`ip address 192.168.1.1/24`
	Add the IP address and mask for interface `vlan2`.

On switch_B, add a different IP interface to virtual router.

`awplus#`
`configure terminal`
	Enter Global Configuration mode.

`awplus(config)#`
`hostname Switch_B`
	Assign a host name to `Switch_B`.

`Switch_B(config)#`
`interface vlan2`
	Specify the interface (`vlan2`) that will participate in virtual routing.

`Switch_B(config)#`
`ip address 192.168.1.2/24`
	Add the IP address and mask for interface `vlan2`.

On switch_C, add a third IP interface to the virtual router.

awplus# `configure terminal`	Enter Global Configuration mode.
awplus(config)# `hostname Switch_C`	Assign a host name to `Switch_C`.
Switch_C(config)# `interface vlan2`	Specify the interface (`vlan2`) that will participate in virtual routing.
Switch_C(config)# `ip address 192.168.1.3/24`	Add the IP address and mask for interface `vlan2`.

Step 2: Create the virtual router

On switch A, create virtual router `vrrp2` with IP address `192.168.1.4`, plaintext authentication with password `trip4e`, and a high priority.

To configure the virtual router on switch_A

Switch_A# `configure terminal`	Enter the Global Configuration mode.
Switch_A(config)# `router vrrp2 vlan2`	Create a new VRRP session on the router, specify the VRID for the session, and specify the interface (`vlan2`) that will participate in virtual routing.
Switch_A(config-router)# `virtual-ip 192.168.1.4 backup`	Set the virtual IP address for the VRRP session. Define the default state of the VRRP router within the virtual router.
Switch_A(config-router)# `preempt-mode on`	Turn on preempt mode.
Switch_A(config-router)# `priority 254`	Set the VRRP priority of 254 for the switch.
Switch_A(config-router)# `enable`	Enable VRRP on the switch.
Switch_A(config-router)# `exit`	Return to the Global Configuration mode.
Switch_A(config)# `interface vlan2`	Specify the interface (`vlan2`) that will participate in virtual routing.
Switch_A(config-if)# `ip vrrp authentication mode text`	Apply text mode authentication to `vlan2`.

To configure the virtual router on switch_A

Switch_A(config-if)# ip vrrp authentication string trip4e	Specify the authentication string trip4e used by the key.
Switch_A(config)# exit	Return to the Privileged Exec mode prompt.
Switch_A#	Privileged Exec mode prompt.

On switch B, create the same virtual router, but with a lower priority.

To configure the virtual router on switch_B

Switch_B# configure terminal	Enter the Global Configuration mode.
Switch_B(config)# router vrrp2 vlan2	Create a new VRRP session on the router, specify the VRID for the session, and specify the interface (vlan2) that will participate in virtual routing.
Switch_B(config-router)# virtual-ip 192.168.1.4 backup	Set the virtual IP address for the VRRP session. Define the default state of the VRRP router within the virtual router.
Switch_B(config-router)# preempt-mode on	Turn on preempt mode.
Switch_B(config-router)# priority 200	Set the VRRP priority of 200 for the switch.
Switch_B(config-router)# enable	Enable VRRP on the switch.
Switch_A(config-router)# exit	Return to the Global Configuration mode.
Switch_B(config)# interface vlan2	Specify the interface (vlan2) that will participate in virtual routing.

To configure the virtual router on switch_B

`Switch_B(config-if)#`
`ip vrrp authentication mode text`
	Apply text mode authentication to `vlan2`.

`Switch_B(config-if)#`
`ip vrrp authentication string trip4e`
	Specify the authentication string `trip4e` used by the key.

`Switch_B(config-if)#`
`exit`
	Return to the Global Configuration mode.

`Switch_B(config)#`
`exit`
	Return to the Privileged Exec mode prompt.

`Switch_B#`	Privileged Exec mode prompt.

On switch C, create the same virtual router with the default priority of 100.

To configure the virtual router on switch_C

`Switch_C#`
`configure terminal`
	Enter the Global Configuration mode.

`Switch_C(config)#`
`router vrrp2 vlan2`
	Create a new VRRP session on the router, specify the VRID for the session, and specify the interface (`vlan2`) that will participate in virtual routing.

`Switch_C(config-router)#`
`virtual-ip 192.168.1.4 backup`
	Set the virtual IP address for the VRRP session. Define the default state (master or backup) of the VRRP router within the virtual router.

`Switch_C(config-router)#`
`preempt-mode on`
	Turn on preempt mode.

`Switch_C(config-router)#`
`priority 100`
	Set the VRRP priority of 100 for the switch.

`Switch_C(config-router)#`
`enable`
	Enable VRRP on the switch.

`Switch_A(config-router)#`
`exit`
	Return to the Global Configuration mode.

`Switch_C(config)#`
`interface vlan2`
	Specify the interface (`vlan2`) that will participate in virtual routing.

`Switch_C(config-if)#`
`ip vrrp authentication mode text`
	Apply text mode authentication to `vlan2`.

`Switch_C(config-if)#`
`ip vrrp authentication string trip4e`
	Specify the authentication string `trip4e` used by the key.

`Switch_B(config-if)#`
`exit`
	Return to the Global Configuration mode.

`Switch_C(config)#`
`exit`
	Return to the Privileged Exec mode prompt

`Switch_C#`	Privileged Exec mode prompt

The default preempt mode ensures that the highest priority switch available always takes the master role. However, if there are no significant disadvantages to the lower priority switches having the master role, and if changes where the switch takes the master role are to be avoided (for example, when a high cost is associated with each change) then you should instead set the preempt mode to **off**.

Chapter 55: VRRP Commands

Command List

This chapter provides an alphabetical reference for commands used to configure the Virtual Router Redundancy Protocol (VRRP). For more information, see Chapter 54, VRRP Introduction and Configuration.

For information about modifying or redirecting the output from **show** commands to a file, see "Controlling "show" Command Output" on page 1.41.

advertisement-interval

Use this command to configure the advertisement interval of the virtual router. This is the length of time, in seconds, between each advertisement sent from the master to its backup(s).

Use the **no** variant of this command to remove an advertisement interval of the virtual router, which has been set using the **advertisement-interval** command.

Syntax

```
advertisement-interval <1-255>

no advertisement-interval
```

Parameter	Description
<1-255>	Specifies the advertisement interval in seconds.

Default The default advertisement interval is 1 second.

Mode Router Configuration

Example

```
               awplus# configure terminal
       awplus(config)# router vrrp 5 vlan2
awplus(config-router)# interface vlan2
awplus(config-router)# advertisement-interval 6
```

```
               awplus# configure terminal
       awplus(config)# router vrrp 5
awplus(config-router)# interface vlan2
awplus(config-router)# no advertisement-interval
```

circuit-failover

Use this command to enable the VRRP circuit failover feature.

Use the **no** variant of this command to disable this feature.

Syntax

```
circuit-failover <interface> <1-253>

no circuit-failover [<interface> <1-253>]
```

Parameter	Description
`<interface>`	The interface of the router that will participate in the virtual router. Interface must exist on the router.
`<1-253>`	Delta value. The value by which virtual routers decrement their priority value during a circuit failover event. Configure this value to be greater than the difference of priorities on the master and backup routers.

Mode Router Configuration

Example

```
              awplus# configure terminal
      awplus(config)# router vrrp 1
awplus(config-router)# interface vlan1
awplus(config-router)# circuit-failover vlan2 30

              awplus# configure terminal
      awplus(config)# router vrrp 1
awplus(config-router)# interface vlan1
awplus(config-router)# no circuit-failover
```

Related Commands router vrrp (interface)

debug vrrp

Use this command to specify debugging options for VRRP. The **all** parameter turns on all the debugging options.

Use the **no** variant of this command to disable this function.

Syntax

```
debug vrrp [all]

no debug vrrp [all]
```

Mode Privileged Exec and Global Configuration

Examples

```
        awplus#  configure terminal
awplus(config)#  debug vrrp all

        awplus#  configure terminal
awplus(config)#  no debug vrrp
```

Related Commands undebug vrrp

debug vrrp events

Use this command to specify debugging options for VRRP event troubleshooting.

Use the **no** variant of this command to disable this function.

Syntax

```
debug vrrp events
```

```
no debug vrrp events
```

Mode Privileged Exec and Global Configuration

Usage The **debug vrrp events** command enables the display of debug information related to VRRP internal events.

Examples

```
        awplus# configure terminal
awplus(config)# debug vrrp events
```

```
        awplus# configure terminal
awplus(config)# no debug vrrp events
```

Related Commands undebug vrrp events

debug vrrp packet

Use this command to specify debugging options for VRRP packets.

Use the **no** variant of this command to disable this function.

Syntax

```
debug vrrp packet [send|recv]
```

```
no debug vrrp packet [send|recv]
```

Parameter	Description
`send`	Specifies the debug option set for sent packets.
`recv`	Specifies the debug option set for received packets.

Mode Privileged Exec and Global Configuration

Usage The **debug vrrp packet** command enables the display of debug information related to the sending and receiving of packets.

Examples

```
        awplus# configure terminal
awplus(config)# debug vrrp packet send
```

```
        awplus# configure terminal
awplus(config)# no debug vrrp packet
```

Related Commands undebug vrrp packet

disable (VRRP)

Use this command to disable a VRRP session on the router to stop it participating in virtual routing.

Syntax

```
disable
```

Mode

Router Configuration

Example

```
              awplus# configure terminal
      awplus(config)# router vrrp 5
awplus(config-router)# interface vlan2
awplus(config-router)# disable
```

Related Commands

enable (VRRP)

enable (VRRP)

Use this command to enable the VRRP session on the router to make it participate in virtual routing.

Syntax

```
enable
```

Mode

Router Configuration

Usage

You must configure the virtual IP address and define the interface for the VRRP session (using the **virtual-ip** and **interface** commands) before using this command.

Example

```
             awplus# configure terminal
     awplus(config)# router vrrp 5
awplus(config-router)# interface vlan2
awplus(config-router)# enable
```

Related Commands

disable (VRRP)
show vrrp

interface (VRRP) (deprecated)

This command has been deprecated and will be removed in a later software version.

Use the router vrrp (interface) command to configure VRRP and define the interface that will participate in virtual routing to send and receive advertisement messages.

Use this command to define the physical interface that will participate in virtual routing. This interface is used for two purposes - to send/receive advertisement messages and to forward on behalf of the virtual router when in master state. Note that you can also specify the interface within the **circuit-failover** and the **router vrrp** commands without specifying the interface later.

Syntax

```
interface <interface>
```

Parameter	Description
<interface>	Specify the name of the interface that will participate in the virtual routing. Interface must exist on the router.

Mode Router Configuration

Usage Use the **no** variant of this command to remove the specified interface from participating in virtual routing. Note that the interface is now specified in the router vrrp (interface) command.

Example To configure an interface on which VRRP is enabled enter the commands shown below:

```
              awplus#  configure terminal
      awplus(config)#  router vrrp 5
awplus(config-router)#  interface vlan2
```

Related Commands router vrrp (interface)

ip vrrp authentication mode

Use this command to enable clear text password authentication used for VRRP packets.

Use the ip vrrp authentication string command after this command to specify the password.

Use the **no** variant of this command to reset to the default of no text authentication.

Syntax

```
ip vrrp authentication mode text
```

```
no ip vrrp authentication mode [text]
```

Parameter	Description
`text`	Specifies the clear text or simple password authentication.

Default No text authentication.

Mode Interface Configuration for a VLAN interface.

Usage RFC 3768 *Virtual Router Redundancy Protocol (VRRP)* recommends no authentication. VRRP authentication commands are available for backwards compatibility the earlier VRRP RFC 2338.

See“VRRP authentication” on page 54.7 for further information about VRRP Authentication.

Examples The following example shows text authentication configured on the `vlan2` interface ensuring authentication packets received on this interface.

```
        awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ip vrrp authentication mode text
```

The following example resets to the default setting for no text authentication on `vlan2`.

```
        awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# no ip vrrp authentication mode
```

Related Commands ip vrrp authentication string

ip vrrp authentication string

Use this command to specify the authentication string or password used by a key.

Use this command after ip vrrp authentication mode that enables clear text authentication.

Use the **no** variant of this command to remove a configured authentication string.

Syntax

```
ip vrrp authentication string <password>

no ip vrrp authentication string
```

Parameter	Description
<password>	The authentication string or password.

Mode Interface Configuration for a VLAN interface.

Example In the following example, the interface `vlan2` is configured to have an authentication string as `guest`, any receiving packet in that interface should have the same string as password.

```
            awplus# configure terminal
    awplus(config)# interface vlan2
 awplus(config-if)# ip vrrp authentication mode text
 awplus(config-if)# ip vrrp authentication string guest
```

See “VRRP authentication” on page 54.7 for further information about VRRP Authentication.

Related Commands ip vrrp authentication mode

preempt-mode

Use this command to configure preempt mode. If set to **true**, the highest priority backup will always be the master when the default master is unavailable. If set to **false**, a higher priority backup will not preempt a lower priority backup who is acting as master.

Syntax

```
preempt-mode {true|false}
```

Parameter	Description
`true`	Preemption enabled.
`false`	Preemption disabled.

Default The default is **true**.

Mode Router Configuration

Usage When the master router fails, the backup routers come online in priority order—highest to lowest. Preempt mode means that a higher priority back up router will take over the master role from a lower priority back up. Preempt mode on **true** allows a higher priority backup router to relieve a lower priority backup router.

See "VRRP election and preempt" on page 54.6 for further information on preempt mode.

Example

```
              awplus#  configure terminal
      awplus(config)#  router vrrp 4
awplus(config-router)#  interface vlan2
awplus(config-router)#  preempt-mode false
```

Related Commands circuit-failover
priority

priority

Use this command to configure the VRRP router priority within the virtual router. The highest priority router is Master (unless preempt-mode is false).

Use the **no** variant of this command to remove the VRRP router priority within the virtual router, which has been set using the **priority** command.

Syntax

```
priority <1-255>

no priority
```

Parameter	Description
<1-255>	The priority. For the master router, use 255 for this parameter; otherwise use any number from the range <1-254>.

Default Defaults for priority are: **master router** `= 255;` **backup** `= 100.`

Mode Router Configuration

Example

```
          awplus# configure terminal
  awplus(config)# router vrrp 3
awplus(config-router)# interface vlan2
awplus(config-router)# priority 101
```

Related Commands circuit-failover
preempt-mode

router vrrp (interface)

Use this command to configure VRRP and define the interface that will participate in virtual routing to send and receive advertisement messages. This command allows you to enter the Router Configuration mode.

Use the **no** variant of this command to remove the VRRP configuration. Disable the VRRP session before using the **no** variant of this command.

Syntax

```
router vrrp <vrid> <interface>

no router vrrp <vrid> <interface>
```

Parameter	Description
<vrid>	*<1-255>* The ID of the virtual router session to create.
<interface>	Specify the name of the interface that will participate in the virtual routing. The interface must exist on the router.

Mode Global Configuration

Usage Use the required *<interface>* placeholder to define the interface that will participate in virtual routing. This interface is used for two purposes - to send/receive advertisement messages and to forward on behalf of the virtual router when in master state.

Example

```
           awplus# configure terminal
   awplus(config)# router vrrp 5 vlan2
awplus(config-router)#

           awplus# configure terminal
   awplus(config)# no router vrrp 5 vlan2
awplus(config-router)#
```

Related Commands circuit-failover
interface (VRRP) (deprecated)

show debugging vrrp

Use this command to display the set VRRP debugging option.

For information on output options, see “Controlling “show” Command Output” on page 1.41.

Syntax

```
show debugging vrrp
```

Mode User Exec and Privileged Exec

Example

```
awplus# show debugging vrrp
```

show running-config router vrrp

Use this command to show the configuration for VRRP.

Note This command is available only if VRRP is enabled.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax `show running-config router vrrp`

Mode Privileged Exec, Global Configuration, Line Configuration, and Interface Configuration.

Example

```
awplus# show running-config router vrrp
```

Output Figure 55-1: Example output from the **show running-config router vrrp** command

```
!
router vrrp 2 vlan2
 circuit-failover vlan1 3
 advertisement-interval 4
!
```

show vrrp

Use this command to display information about all VRRP sessions. This command shows a summary when the optional **brief** parameter is used.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax

```
show vrrp [brief]
```

Parameter	Description
`brief`	Brief summary of VRRP sessions.

Mode User Exec and Privileged Exec

Example To display information about all VRRP sessions, enter the command:

```
awplus# show vrrp
```

To display brief summary output about VRRP sessions, enter the command:

```
awplus# show vrrp brief
```

Output Figure 55-2: Example output from the **show vrrp** command

```
awplus#show vrrp
VrId <1>
 State is Master
 Virtual IP is 10.0.0.222 (Not IP owner)
 Interface is vlan2
 Priority is 100
 Advertisement interval is 1 sec
 Preempt mode is TRUE
```

Figure 55-3: Example output from the **show vrrp brief** command

```
awplus#show vrrp brief
Interface        Grp  Prio Own  Pre  State      Master addr       Group addr
 vlan10            1   200   N    P  Master     192.168.10.4      192.168.10.253
 vlan10            2   150   N    P  Backup     192.168.10.4      192.168.10.254
 vlan11            3   200   N    P  Master     192.168.11.4      192.168.11.253
 vlan11            4   150   N    P  Backup     192.168.11.4      192.168.11.254
```

Related Commands enable (VRRP)

show vrrp counters

This command displays VRRP SNMP counters on the console, as described in the VRRP MIB and RFC2787, for debugging use while you configure VRRP with commands in this chapter.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax `show vrrp counters`

Mode User Exec and Privileged Exec

Usage The output has a section for global counters and a section of counters for each VRRP instance configured. See the descriptions of the counters below the sample output as per RFC2787.

Note Note that the counters displayed with this commands are the same counters as described in RFC 2787 (Copyright (C) The Internet Society (2000). All Rights Reserved) except for the `Monitored Circuit Up` and `Monitored Circuit Down` counters which are additions beyond the MIB.

Example To display information about VRRP SNMP counters on the console, enter the command:

```
awplus# show vrrp counters
```

Figure 55-4: Example output from the **show vrrp counters** command

```
awplus#show vrrp counters
VRRP Global Counters:
  Checksum Errors .... 230
  Version Errors ..... 0
  VRID Errors ........ 230

VRRP IPv4 counters for VR 10/vlan10:
  Master Transitions ........ 0
  Received Advertisements ... 0
  Internal Errors ........... 0
  TTL Errors ................ 0
  Received Priority 0 Pkt ... 0
  Sent Priority 0 Pkt ....... 0
  Received Invalid Type ..... 0
  Address List Errors ....... 0
  Packet Length Errors ...... 0
  Invalid Authentications ... 0
  Authentication Mismatch ... 0
  Authentication Failures ... 0
  Monitored Circuit Up ...... 0
  Monitored Circuit Down..... 0

VRRP IPv4 counters for VR 100/vlan100:
  Master Transitions ........ 1
  Received Advertisements ... 1614
  Internal Errors ........... 0
  TTL Errors ................ 0
  Received Priority 0 Pkt ... 0
  Sent Priority 0 Pkt ....... 0
  Received Invalid Type ..... 0
  Address List Errors ....... 0
  Packet Length Errors ...... 0
  Invalid Authentications ... 0
  Authentication Mismatch ... 0
  Authentication Failures ... 0
  Monitored Circuit Up ...... 0
  Monitored Circuit Down..... 2
```

Table 55-1: Global counters with descriptions for the **show vrrp counters** command:

Counter	Description
`Checksum Errors`	The total number of VRRP packets received with an invalid VRRP checksum value.
`Version Errors`	The total number of VRRP packets received with an unknown or unsupported version number.
`VRID Errors`	The total number of VRRP packets received with an invalid VRID for this virtual router.

Table 55-2: Per VR counters with descriptions for the **show vrrp counters** command:

Counter	Description
`Master Transitions`	The total number of times that this virtual router's state has transitioned to MASTER.
`Received Advertisements`	The total number of VRRP advertisements received by this virtual router.
`Internal Errors`	The total number of VRRP advertisement packets received for which the advertisement interval is different than the one configured for the local virtual router.
`TTL Errors`	The total number of VRRP packets received by the virtual router with IP TTL (Time-To-Live) not equal to 255.
`Received Priority 0 Pkt`	The total number of VRRP packets received by the virtual router with a priority of '`0`'.
`Sent Priority 0 Pkt`	The total number of VRRP packets sent by the virtual router with a priority of '`0`'.
`Received Invalid Type`	The number of VRRP packets received by the virtual router with an invalid value in the '`type`' field.
`Address List Errors`	The total number of packets received for which the address list does not match the locally configured list for the virtual router.
`Packet Length Errors`	The total number of packets received with a packet length less than the length of the VRRP header.
`Invalid Authentications`	The total number of packets received with an unknown authentication type.
`Authentication Mismatch`	The total number of packets received with '`Auth Type`' not equal to the locally configured authentication method
`Authentication Failures`	The total number of VRRP packets received that do not pass the authentication check.
`Monitored Circuit Up`	The total number of times the monitored circuit has generated the UP event.
`Monitored Circuit Down`	The total number of times the monitored circuit has generated the down event.

show vrrp (session)

Use this command to display information for a particular VRRP session.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax `show vrrp <vrid> <interface>`

Parameter	Description
<vrid>	*<1-255>* The virtual router ID for which to display information. Session must already exist.
<interface>	The interface to display information about, for instance, `vlan2`.

Mode User Exec and Privileged Exec

Usage See the below sample output from the **show vrrp** command displaying information about VRRP session 1 configured on **vlan2**. Output shows that a Virtual IP address has been set.

```
awplus# show vrrp 1 vlan2
```

```
awplus#show vrrp 1 vlan2
Address family IPv4
VrId <1>
 Interface is vlan2
 State is Initialize
 Virtual IP address is 10.10.11.250 (Not IP owner)
 Priority is 100
 Advertisement interval is 1 sec
 Preempt mode is TRUE
```

See the below sample output from the **show vrrp** command displaying information about VRRP session 1 configured on **vlan3**. Output shows a Virtual IP address has not been set yet.

```
awplus# show vrrp 1 vlan3
```

```
awplus#show vrrp 1 vlan3
Address family IPv4
VrId <1>
 Interface is vlan3
 State is Initialize
 Virtual IP address is unset
 Priority is 100
 Advertisement interval is 1 sec
 Preempt mode is TRUE
```

Example The following command shows information about VRRP session 5 for interface **vlan2**.

```
awplus# show vrrp 5 vlan2
```

undebug vrrp

Use this command to disable all VRRP debugging.

Syntax

```
undebug vrrp all
```

Mode Privileged Exec

Example

```
awplus# undebug vrrp all
```

Related Commands debug vrrp

undebug vrrp events

Use this command to disable debugging options for VRRP event troubleshooting.

Syntax

```
undebug vrrp events
```

Mode Privileged Exec

Example

```
awplus# undebug vrrp events
```

Related Commands debug vrrp events

undebug vrrp packet

Use this command to disable debugging options for VRRP packets.

Syntax

```
undebug vrrp packet [send|recv]
```

Parameter	Description
`send`	Disable the debug option set for sent packets.
`recv`	Disable the debug option set for received packets.

Mode Privileged Exec

Example

```
awplus# undebug vrrp packet send
```

Related Commands debug vrrp packet

virtual-ip

Use this command to set the virtual IP address for the VRRP session. This is the IP address of the virtual router that end hosts set as their default gateway.

Use the **no** variant of this command to disable this feature.

Syntax

```
virtual-ip <ip-address> master

virtual-ip <ip-address> backup

no virtual-ip
```

Parameter	Description
<ip-address>	The virtual IP address of the virtual router, entered in the format A.B.C.D.
master	Sets the default state of the VRRP router within the Virtual Router as **master**. For master, the router must own the Virtual IP address.
backup	Sets the default state of the VRRP router within the Virtual Router as **backup**.

Mode Router Configuration

Example

```
               awplus# configure terminal
       awplus(config)# router vrrp 5
awplus(config-router)# interface vlan2
awplus(config-router)# virtual-ip 192.0.2.30 master
```

vrrp vmac

Use this command to enable or disable the Virtual MAC feature.

Syntax

```
vrrp vmac {enable|disable}
```

Mode Global Configuration

Example To enable Virtual MAC enter:

```
        awplus# configure terminal
awplus(config)# vrrp vmac enable
```

To disable Virtual MAC enter:

```
        awplus# configure terminal
awplus(config)# vrrp vmac disable
```

Chapter 56: EPSR Introduction and Configuration

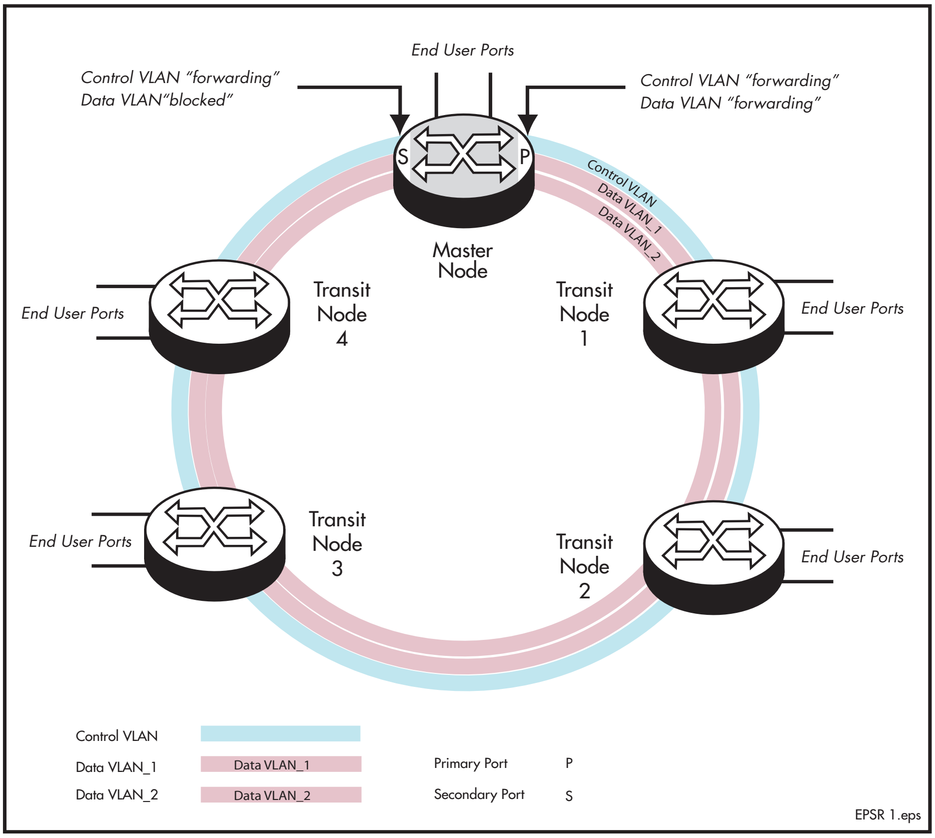
Introduction

Ethernet Protection Switching Ring (EPSR) is a protection system that prevents loops within Ethernet ring based topologies. EPSR offers a rapid detection and recovery time (in the order of 50 ms, depending on configuration) if a link or node fails. This rapid recovery time makes EPSR a more effective alternative to spanning tree options when using ring-based topologies to create high speed resilient Layer 2 networks.

Ring Components and Operation

EPSR operates only on ring-based topologies. An EPSR ring comprises a series of nodes (Ethernet bridges) connected end to end. The figure below shows a basic ring configuration. A ring comprises one master node and a number of transit nodes. Each node connects to the ring via two ports. On the master node one port is configured to be the primary port and the other, the secondary port.

Figure 56-1: Simple EPSR ring configuration



EPSR instances and domains

Each physical EPSR ring contains one or more EPSR domains. An EPSR instance can be thought of as a component of an EPSR ring domain that exists on a single node. A set of instances across the whole ring is called a "domain." Therefore a ring whose individual nodes each have two instances results in a two domain ring. Each instance contains a control VLAN and a number of data VLANs.

The EPSR control VLAN and its associated data VLANs form a Ring Domain. Although a physical ring can have more than one domain, each domain must operate as a separate logical group of VLANs and must have its own master node. This means that several domains may share the same physical network, but must operate as logically separate VLAN groups.

Control VLAN The function of the control VLAN is to monitor the ring domain and maintain its operational functions. To do this it transmits and monitors operational healthcheck messages using EPSR healthcheck control frames. The control VLAN carries no user data.

Data VLAN The data VLAN carries the user data around the ring. Several data VLANs can share a common control VLAN.

Master node The master node controls the ring operation. It issues healthcheck messages at regular intervals from its primary port and monitors their arrival back at its secondary port - after they have circled the ring. Under normal operating conditions the master node's secondary port is always in the blocking state to all data VLAN traffic. This is to prevent data loops forming within the ring. This port however, operates in the forwarding state for the traffic on the control VLAN. Loops do not occur on the control VLAN because the control messages stop at the secondary port, having completed their path around the ring.

Transit nodes The transit nodes operate as conventional Ethernet bridges, but with the additional capability of running the EPSR protocol. This protocol requires the transit nodes to forward the healthcheck messages from the master node, and respond appropriately when a ring fault is detected. The fault condition procedure is explained in "Fault Detection and Recovery" on page 56.4.

Fault Detection and Recovery

EPSR uses the following methods to detect outages in a node or a link in the ring:

- Master node polling fault detection
- Transit node unsolicited fault detection

Master node polling

The master node issues healthcheck messages from its primary port as a means of checking the condition of the EPSR network ring. These messages are sent at regular periods, controlled by the **hellotime** parameter of the epsr command on page 57.4. A failover timer is set each time a healthcheck message leaves the master node's primary port. The timeout value for this timer is set by the **failover** parameter of the epsr command on page 57.4. If the failover timer expires before the transmitted healthcheck message is received by the master node's secondary port, the master node assumes that there is a fault in the ring, and implements its fault recovery procedures. Because this method relies on a timer expiry, its operation is inherently slower than the "transit node unsolicited detection method" described next.

Transit node unsolicited

Transit node unsolicited fault detection relies on transit nodes detecting faults at their interfaces, and immediately notifying master nodes about the break. When a transit node detects a connectivity loss, it sends a "links down" message over its good link. Because a link spans two nodes, both nodes send the "links down" message back to the master node. These nodes also change their state from "links up" to "links down," and change the state of the port connecting to the broken link, from "forwarding" to "blocking."

Fault Recovery

When the master node detects an outage in the ring by using its detection methods, it does the following:

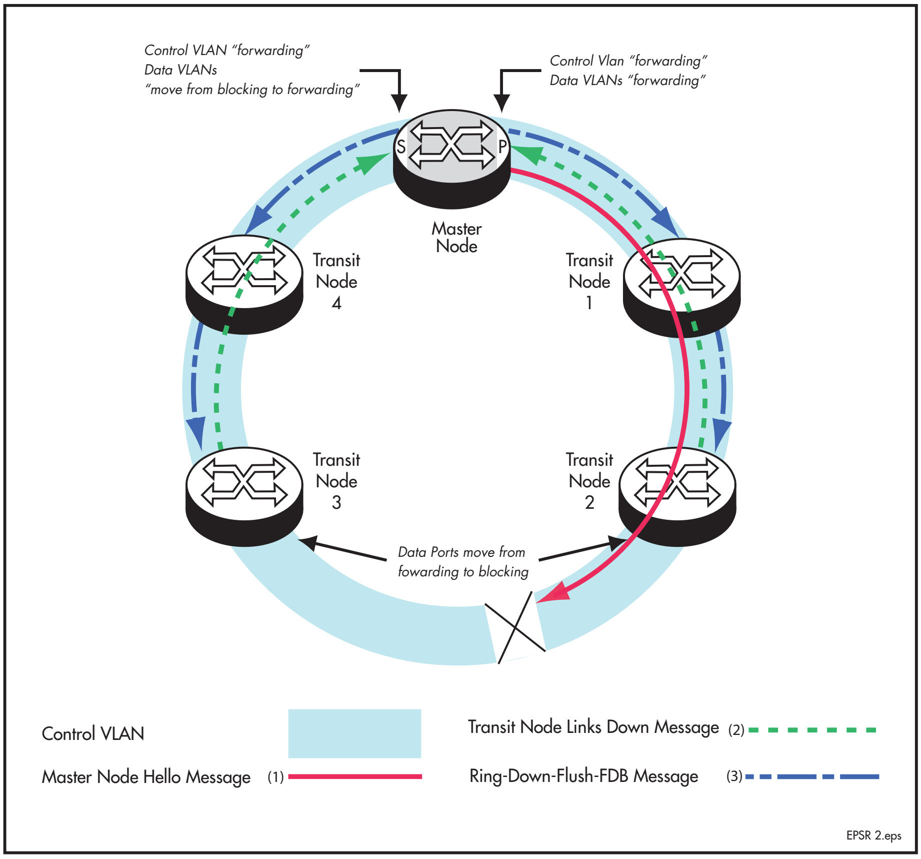
1. Declares the ring to be in a "failed" state.
2. Unblocks its secondary port to enable the data VLAN traffic to pass between its primary and secondary ports.
3. Flushes its own forwarding database (FDB) for (only) the two ring ports.
4. Sends an EPSR Ring-Down-Flush-FDB control message to all the transit nodes, via both its primary and secondary ports.

Transit nodes respond to the Ring-Down-Flush-FDB message by flushing their forward databases for each of their ring ports. As the data starts to flow in the ring's new configuration, each of the nodes (master and transit) re-learn their Layer 2 addresses. During this period, the master node continues to send health check messages over the control VLAN. This situation continues until the faulty link or node is repaired. For a multi-domain ring, this process occurs separately for each domain within the ring.

Note When VCStack is used with EPSR, the EPSR **failovertime** must be set to at least 5 seconds to avoid any broadcast storms during failover. Broadcast storms may occur if the switch cannot failover quickly enough before the EPSR **failovertime** expires. See the epsr command for further information about the EPSR **failovertime**. See the reboot rolling command for further information about VCStack failover.

The following figure shows the flow of control frames under fault conditions.

Figure 56-2: EPSR Fault Detection Messages


Control VLAN "forwarding"
Data VLANs
"move from blocking to forwarding"
Control Vlan "forwarding"
Data VLANs "forwarding"
S
P
Master
Node
Transit
Node
4
Transit
Node
1
Transit
Node
3
Transit
Node
2
Data Ports move from
fowarding to blocking
Control VLAN
Transit Node Links Down Message (2)
Master Node Hello Message (1)
Ring-Down-Flush-FDB Message (3)
EPSR 2.eps

Restoring Normal Operation

Transit nodes

Once a fault in the ring or node has been rectified, the transit nodes that span the previously faulty link section detect that link connectivity has returned. They then move their appropriate ring port state, from Links-Down to Pre-Forwarding, and await the Ring-Up-Flush control message from the master node.

Once these transit nodes receive the Ring-Up-Flush message, they:

- flush their forward databases for both their ring ports.
- change the state of their ports from blocking to forwarding, which allows data to flow through their previously blocked ring ports.

Note The transit nodes do not enter the forward state until they have received the Ring-Up-Flush message. This prevents the possibility of a loop condition occurring caused by the transit nodes moving into the forwarding state before the master node secondary port can return to the blocking state. During such a period, the ring would have no ports blocked.

Master node

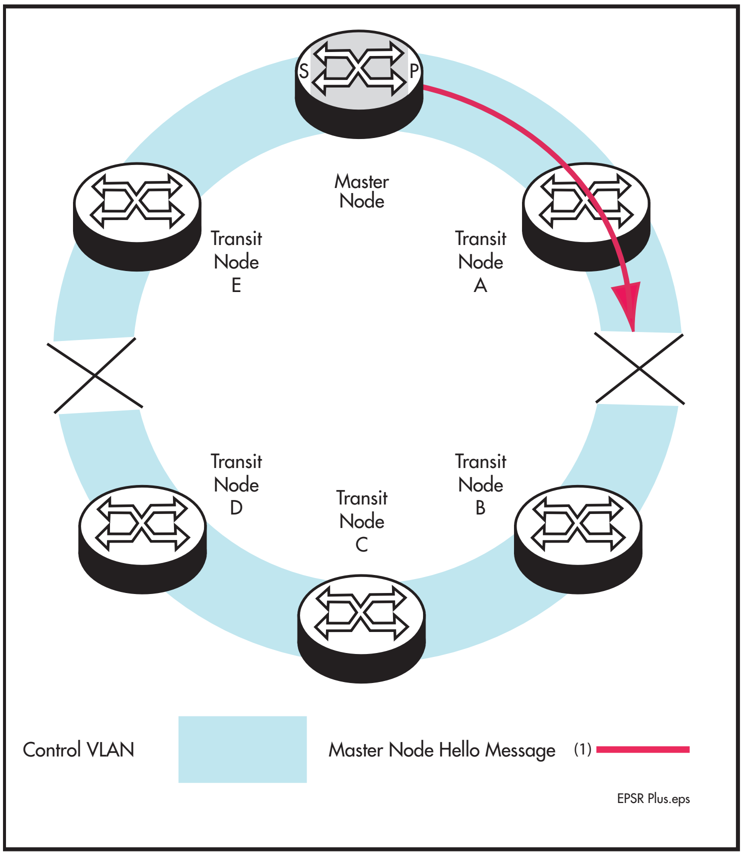
With the link restored, the healthcheck messages that are sent from the primary port of the master node now complete the loop and arrive at the master node's secondary port. The master node restores normal conditions as follows:

1. Declares the ring to be in a “complete” state.
2. Blocks its secondary port for data (non-control) traffic.
3. Flushes its forwarding database for its two ring ports.
4. Sends a Ring-Up-Flush-FDB message from its primary port, to all transit nodes.

Managing Rings with Two Breaks

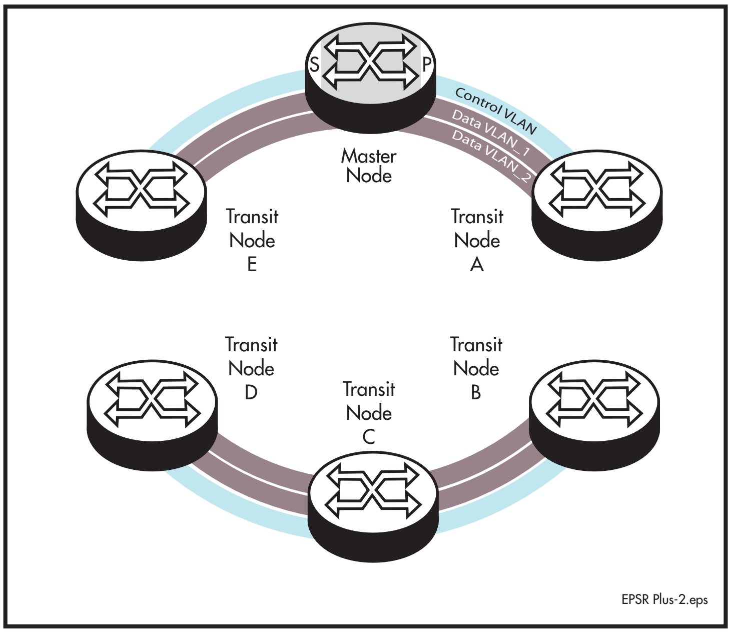
To restore a link with two breaks you need to run the EPSR Enhanced Recovery feature. Consider the network shown below:

Figure 56-3: EPSR Ring with Two Breaks



In this situation the ring will attempt to recover as previously described in "Fault Recovery" on page 56.4. This will result in the split-ring operation shown in Figure 56-4 on page 56.7.

Figure 56-4: EPSR Split Ring

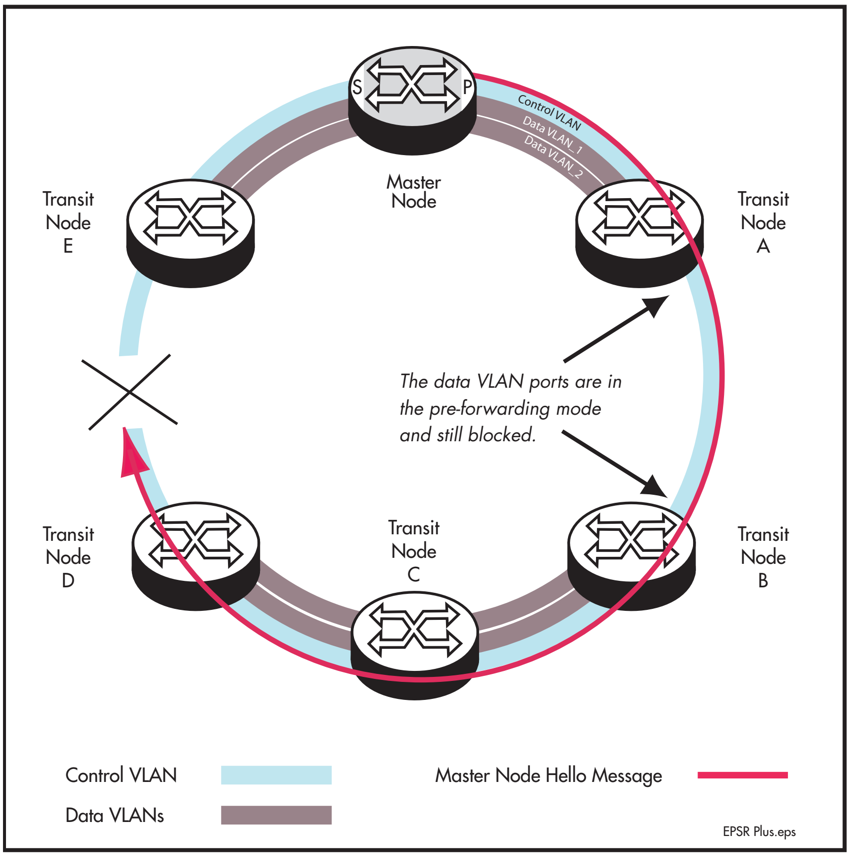


In this operational mode each portion of the ring operates as an independent link layer broadcast domain each containing the original data VLANs and control VLAN.

Recovery When One Break is Restored

Figure 56-5 on page 56.8 shows a ring with the link between nodes A and B restored. At this point the ring's behavior will depend on whether the epsr enhancedrecovery enable command on page 57.8 has been set.

Figure 56-5: EPSR Ring with One Link Restored



The data VLAN ports are in the pre-forwarding mode and still blocked.

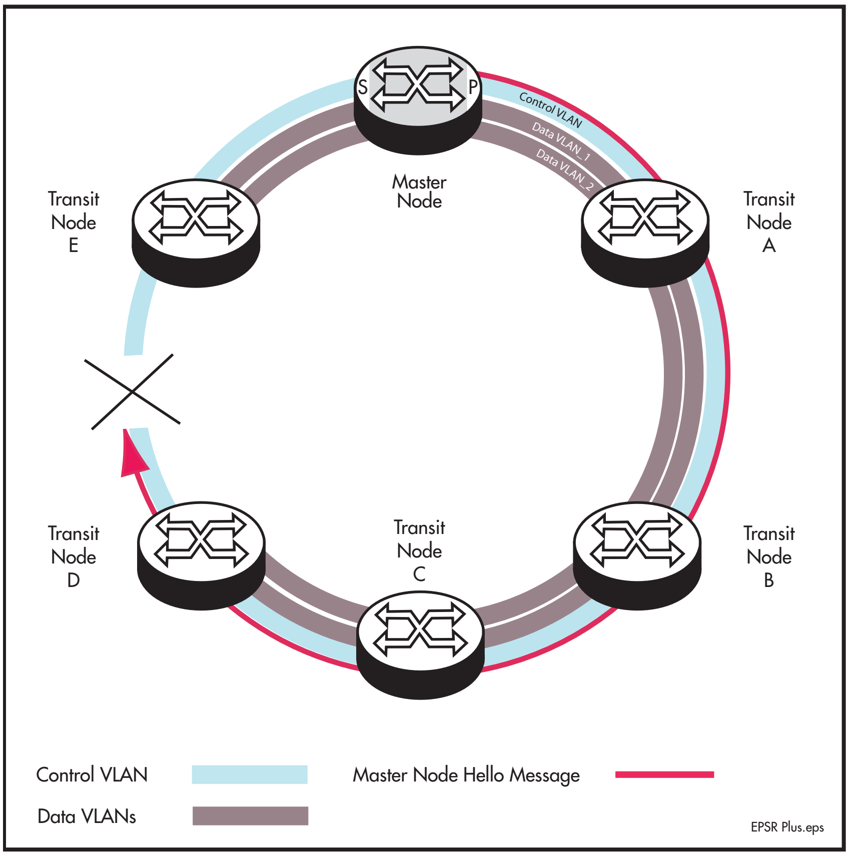
Enhanced Recovery Disabled

With the enhanced recovery feature disabled, the Hello messages will now reach the remaining ring break; however from a users perspective, the ring will remain as shown in the split state shown in Figure 56-4.

Enhanced Recovery Enabled

With the enhanced recovery feature enabled, switch nodes A and B are able to detect the restored link, and will place all their ring ports in the forwarding state. Although the ring will remain in the "failed" state because of the remaining break; communication between the nodes is restored. The network then operates as shown in Figure 56-6.

Figure 56-6: EPSR Operation in Partially Recovered State



Configuration Examples

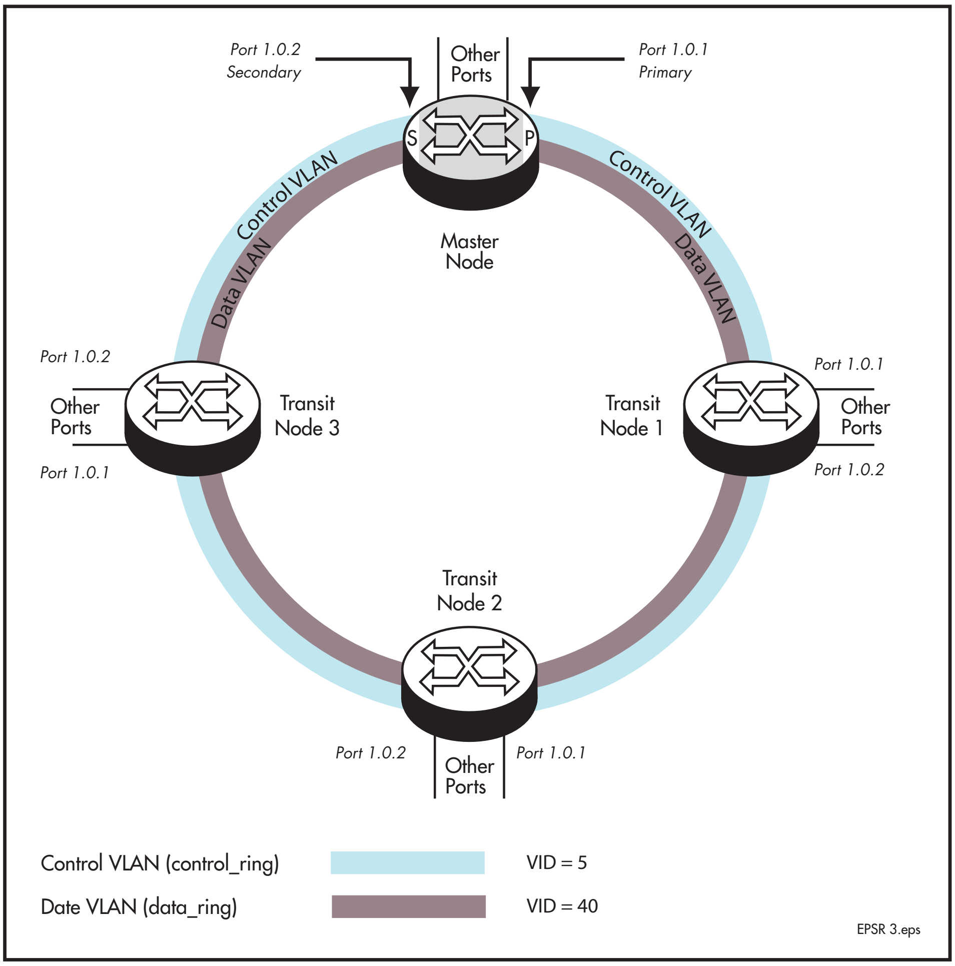
This section describes how to configure EPSR in following ways:

- Single Domain, Single Ring Network
- Single Ring, Dual Domain Network
- EPSR and Spanning Tree Operation

Single Domain, Single Ring Network

This example shows a simple single ring, single domain configuration with no connecting lobes.

Figure 56-7: EPSR single domain, single ring network



Configure the Master Node

Step 1: Create the control and data VLANs on the Master Node

`awplus#`
`configure terminal`
	Enter the Global Configuration mode.

`awplus(config)#`
`vlan database`
	Enter the VLAN Configuration mode.

`awplus(config-vlan)#`
`vlan 5 name control_vlan state enable`
	Enable VLAN 5 called `control_vlan` on the Master Node. Specifying the enable state allows forwarding of frames on the VLAN-aware node.

`awplus(config-vlan)#`
`vlan 40 name data_vlan state enable`
	Enable VLAN 40 called `data_vlan` on the Master Node. Specifying the enable state allows forwarding of frames on the VLAN-aware node.

`awplus(config-vlan)#`
`exit`
	Exit the VLAN Configuration mode and enter the Global Configuration mode.

Step 2: Add port1.0.1 to these VLANs

`awplus(config)#`
`interface port1.0.1`
	Specify the interface (`port1.0.1`) that you are configuring and enter the Interface Configuration mode.

`awplus(config-if)#`
`switchport mode trunk`
	Set the switching characteristics of this port to Trunk mode.

`awplus(config-if)#`
`switchport trunk allowed vlan add 5`
	Enable VLAN 5 on this port.

`awplus(config-if)#`
`switchport trunk allowed vlan add 40`
	Enable VLAN 40 on this port.

`awplus(config-if)#`
`exit`
	Exit the Interface mode and enter the Global Configuration mode.

Step 3: Add port1.0.2 to these VLANs

awplus(config)# interface port1.0.2	Specify the interface (port1.0.2) that you are configuring and enter the Interface Configuration mode.
awplus(config-if)# switchport mode trunk	Set the switching characteristics of this port to Trunk mode.
awplus(config-if)# switchport trunk allowed vlan add 5	Enable VLAN 5 on this port.
awplus(config-if)# switchport trunk allowed vlan add 40	Enable VLAN 40 on this port.
awplus(config-if)# exit	Exit the Interface Configuration mode and enter the Global Configuration mode.

Step 4: Create the EPSR Instance called “blue” on the master node, make VLAN 5 the control VLAN and port 1.0.1 the primary port

awplus(config)# epsr configuration	Enter the EPSR Configuration mode.
awplus(config-epsr)# epsr blue mode master controlvlan 5 primaryport port1.0.1	Create an EPSR instance called blue on vlan5. Make vlan5 the control VLAN. Make port 1.0.1 the primary port. Make this node the master.

Step 5: Add a data VLAN to the EPSR Instance called “blue” on the Master Node

awplus(config-epsr)# epsr blue datavlan 40	On epsr instance called blue make vlan40 the data VLAN.

Step 6: Enable the EPSR Instance called “blue” on the Master Node

awplus(config-epsr)# epsr blue state enable	Enable the EPSR instance named blue.
awplus(config-epsr)# exit	Exit the EPSR Configuration mode.

Now you can configure the transit nodes.

Step 7: Create the Control and Data VLANs on a Transit Node

awplus# `configure terminal`	Enter the Global Configuration mode.
awplus(config)# `vlan database`	Enter the VLAN Configuration mode.
awplus(config-vlan)# `vlan 5 name control_vlan state enable`	Enable VLAN 5 called `control_vlan` on the Transit Node. Specifying the enable state allows forwarding of frames on the VLAN-aware node.
awplus(config-vlan)# `vlan 40 name data_vlan state enable`	Enable VLAN 40 called `data_vlan` on the Transit Node. Specifying the enable state allows forwarding of frames on the VLAN-aware node.
awplus(config-vlan)# `exit`	Exit the VLAN Configuration mode and enter the Global Configuration mode.

Step 8: Add port1.0.1 to the VLANs

awplus(config)# `interface port1.0.1`	Specify the interface (`port1.0.1`) that you are configuring and enter the Interface Configuration mode.
awplus(config-if)# `switchport mode trunk`	Set the switching characteristics of this port to Trunk mode.
awplus(config-if)# `switchport trunk allowed vlan add 5`	Enable VLAN 5 on this port.
awplus(config-if)# `switchport trunk allowed vlan add 40`	Enable VLAN 40 on this port.
awplus(config-if)# `exit`	Exit the Interface Configuration mode and enter the Global Configuration mode.

Step 9: Add port1.0.2 to the VLANs

awplus(config)# interface port1.0.2	Specify the interface (port1.0.2) that you are configuring and enter the Interface Configuration mode.
awplus(config-if)# switchport mode trunk	Set the switching characteristics of this port to Trunk mode.
awplus(config-if)# switchport trunk allowed vlan add 5	Enable VLAN 5 on this port.
awplus(config-if)# switchport trunk allowed vlan add 40	Enable VLAN 40 on this port.
awplus(config-if)# exit	Exit the Interface Configuration mode and enter the Global Configuration mode.

Step 10: Create the EPSR Instance called “blue” on a transit node, make VLAN 5 the control VLAN

awplus(config)# epsr configuration	Enter the EPSR Configuration mode.
awplus(config-epsr)# epsr blue mode transit controlvlan 5	Create an EPSR instance called blue on vlan5. Make vlan5 the control VLAN. Make this node a transit node.

Step 11: Add a data VLAN to the EPSR Instance called “blue” on the transit node

awplus(config-epsr)# epsr blue datavlan 40	On the EPSR instance called blue make vlan40 the data VLAN.

Step 12: Enable the EPSR Instance called “blue” on the transit node

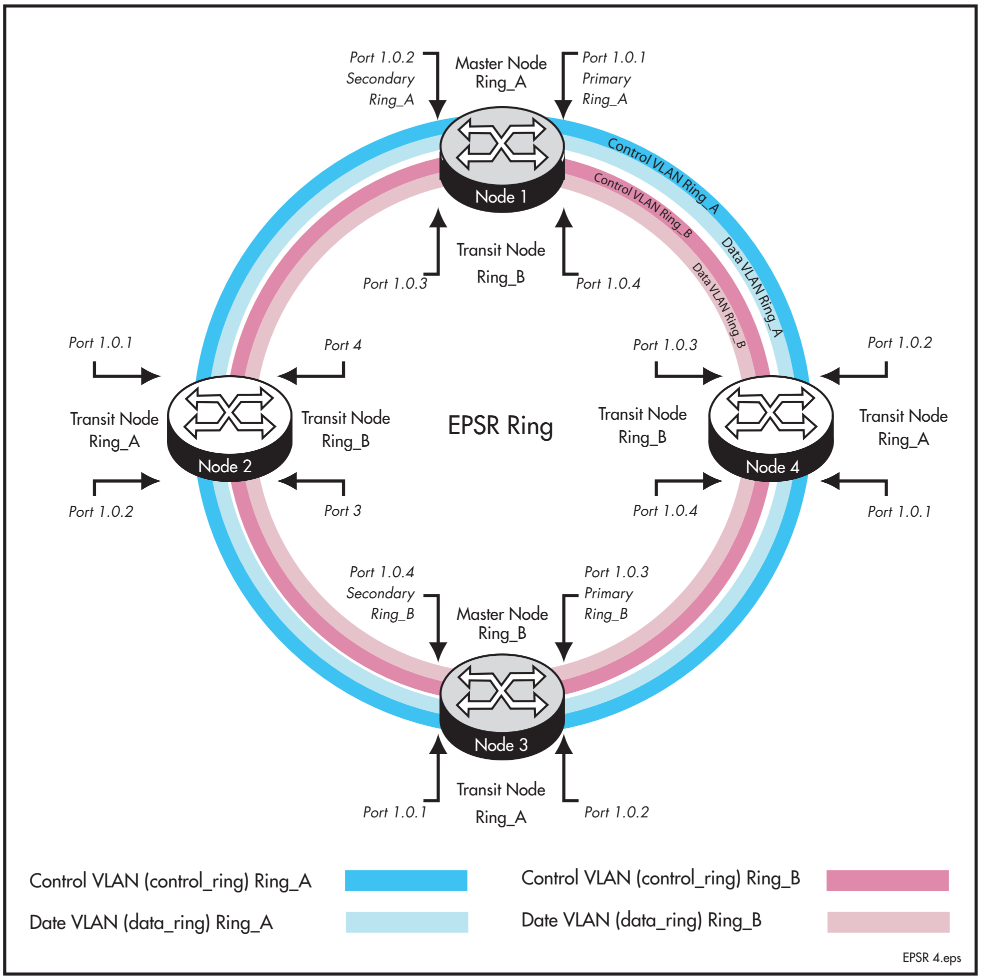
awplus(config-epsr)# epsr blue state enable	Enable the EPSR instance named blue.
awplus(config-epsr)# exit	Exit the EPSR Configuration mode.

Now you can use the same procedure to configure the remaining transit nodes.

Single Ring, Dual Domain Network

This example shows an EPSR configuration where two EPSR domains share the same physical ring. This configuration enables two sets of users to run totally separate Layer 2 networks. Better load distribution around the ring can be achieved by configuring different nodes to be the master for each ring.

Figure 56-8: EPSR single ring network, two domain network.

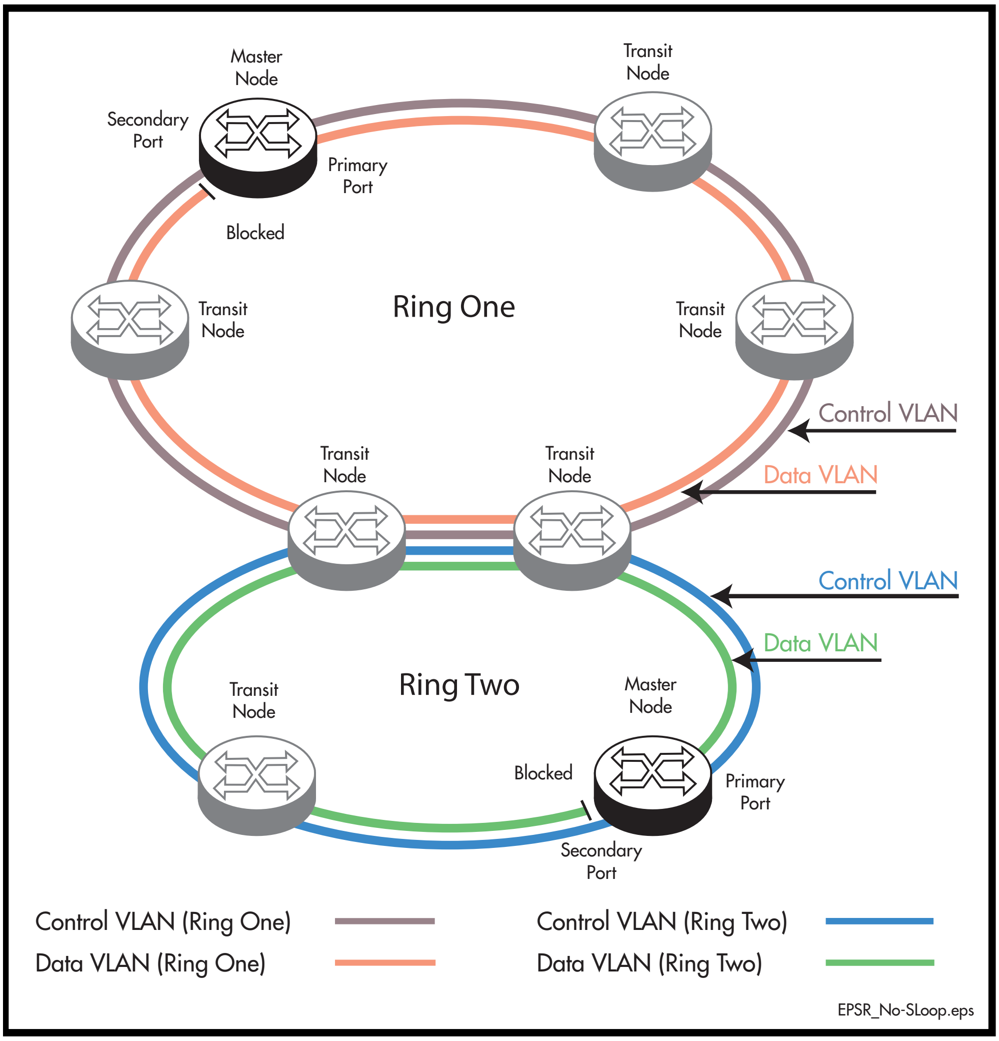


Interconnected Rings

This example shows an EPSR configuration where two rings share a common segment. This configuration will operate as two independent rings, providing that there is no data VLAN sharing between the two rings. If a break occurs in either ring then, each ring will implement its own independent recovery procedures. If a break occurs in the common segment, then each Master node will unblock its secondary port using the normal fault recovery procedure.

Where data VLANS are shared between the rings a fault condition know as "SuperLoop" can occur. The next section deals with superloops and how to manage them.

Figure 56-9: Interconnected EPSR Rings with No Data VLAN Sharing



Superloop Protection

Careful attention must be paid when creating EPSR networks with interconnecting links, to avoid an error condition known as superloops. This sections explains what superloops are and how to prevent them.

What is a an EPSR Superloop?

An EPSR superloop is a data loop whose path traverses more than a single EPSR ring. This fault condition usually occurs when there is a break in a physical segment that is shared by the two rings. For a superloop condition to occur, the two physical rings must share some of their data VLANS. Figure 56-10 on page 56.17shows an EPSR ring with a superloop condition caused by a break in the common ring segment. Figure 56-11 on page 56.18 shows the Superloop data path ring caused by the broken common ring segment. The superloop condition occurs because both rings detect the ring segment break and as a result both master nodes unblock their secondary ports.

Figure 56-10: Interconnected EPSR Rings with Data VLAN Sharing

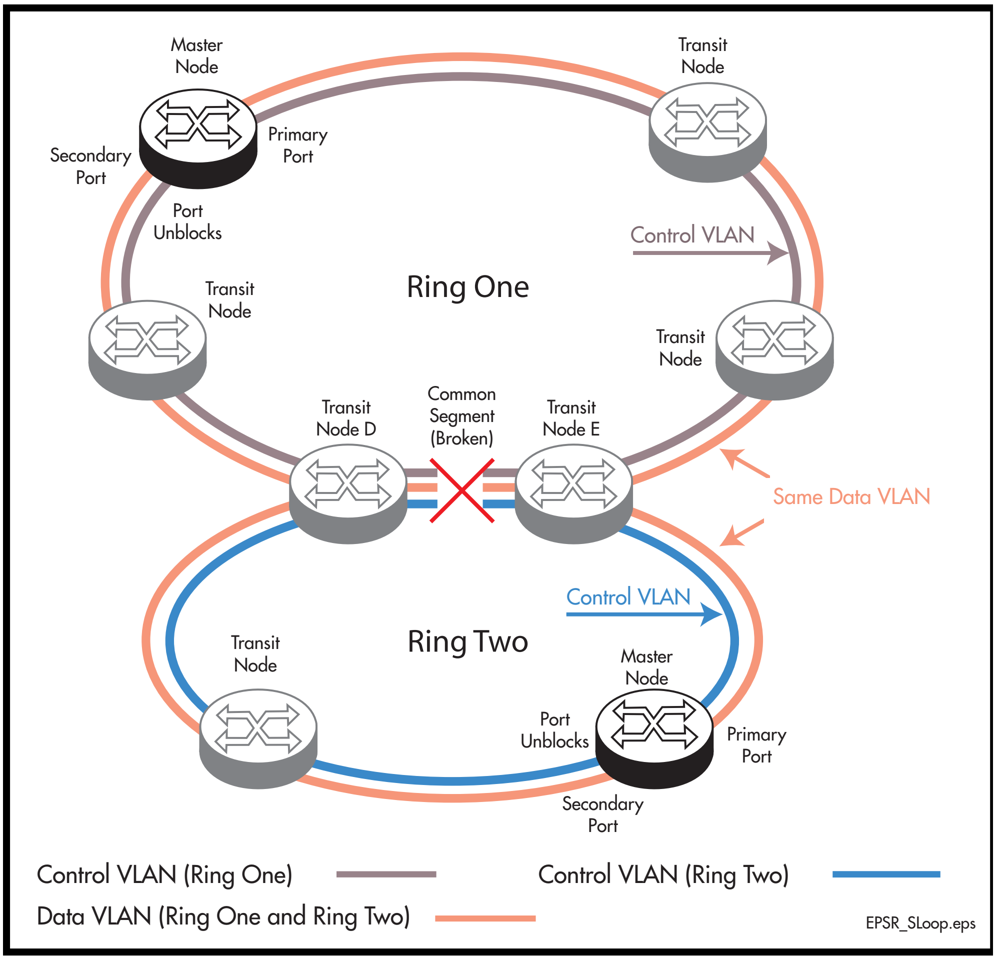
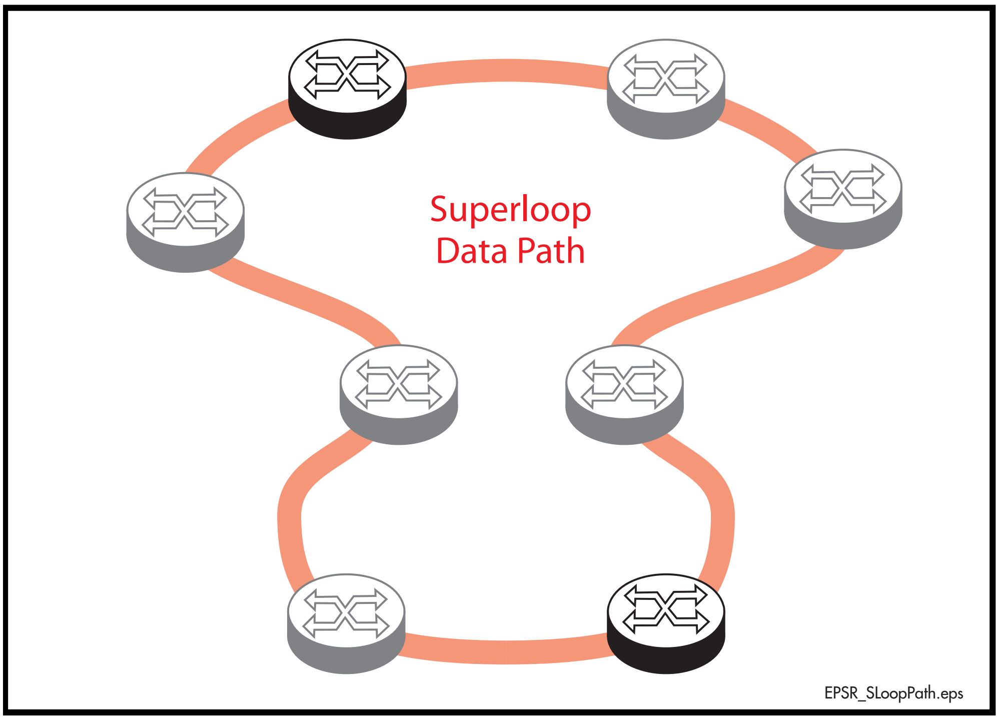


Figure 56-11: EPSR Superloop data path caused by a broken common ring segment



EPSR Superloop Prevention

Alliedware Plus version 5.4.2 onwards contains mechanisms to prevent superloops forming. The Superloop prevention facility enables rings to be assigned priority level between 0 and 127, with 1 representing the lowest priority and 127 the highest. Level 0 (the default setting) applies the functionality of no Superloop prevention. Enabling superloop prevention changes the way the EPSR nodes respond under fault conditions.

Superloop prevention is enabled for an EPSR ring instance by setting the **epsr priority** command on page 57.11. Setting a priority value greater than 0 applies superloop prevention to that particular instance. How the superloop function is applied will depend on the role of the node within the ring, i.e. whether it is a master node or a transit node, and its physical location within the ring. Here is how the functions of Superloop prevention modify the nodal behavior for a particular ring instance:

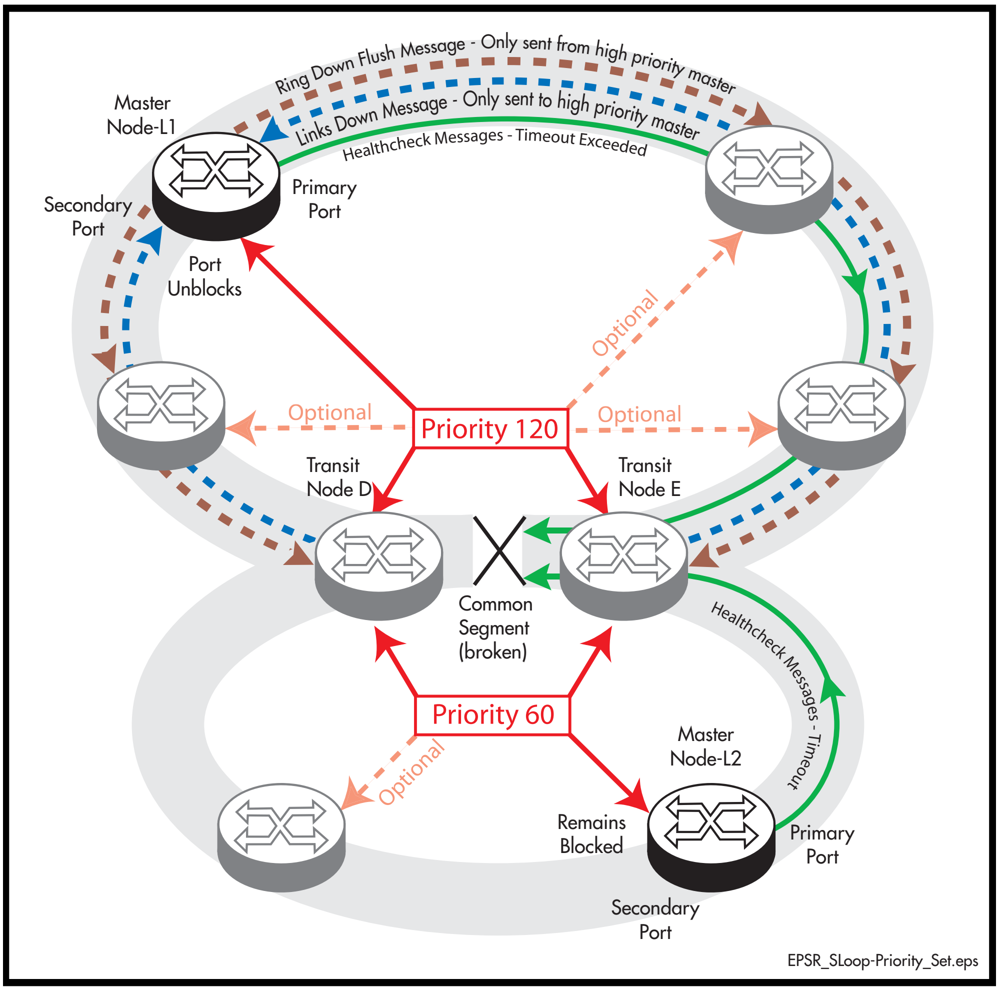
- A master node with its epsr priority set to zero will consider the superloop function to be turned off.
- A master node with its epsr priority set within the range 1-127 will consider the superloop function to be enabled, and will change its behavior in the following ways.
  - It will **not** unblock its secondary port following the expiry of the Master Node Hello message timer. However, a ring-down-flush message will still be sent.
  - It **will** only unblock its secondary port when it receives a Links Down message from a transit node.
- A transit node that is not connect to a shared link will be unaffected by having its epsr priority set for any of its instances.

- A transit node that is connected to a shared link will change its behavior in the following ways:
  - « It will compare its priority settings applied to each of the instances sharing the common link. So for the network of Figure 56-10 on page 56.17 Transit Node D will compare the priority setting for Ring One, with the priority setting for Ring Two.

    If the shared link fails, the transit node will only issue a **Transit Node Links Down message** on the ring that is configured with the highest priority.

The result of these behavior changes is that when the shared link fails, only the master node located on the higher priority ring will unblock its secondary port; because this is the only the master node that will receive the **Transit Node Links Down message**. Note also that the master node will receive these messages from the transit nodes at either end of the broken shared link (Nodes D and E). This concept is illustrated in

Figure 56-12: EPSR behavior under fault conditions with Superloop enabled


Ring Down Flush Message - Only sent from high priority master
Links Down Message - Only sent to high priority master
Healthcheck Messages - Timeout Exceeded
Master
Node-L1
Secondary
Port
Primary
Port
Port
Unblocks
Optional
Optional
Priority 120
Optional
Transit
Node D
Transit
Node E
Common
Segment
(broken)
Healthcheck Messages - Timeout
Priority 60
Optional
Master
Node-L2
Remains
Blocked
Primary
Port
Secondary
Port
EPSR_SLoop-Priority_Set.eps

For this process to work requires certain configuration rules to be obeyed.

Configuration Rules for Superloop Protected EPSR Rings

The following configuration rules are advised when configuring EPSR rings that share one or more common segments.

- Allocate a priority order to each of the interconnected rings, with 127 being the highest priority and 1 the lowest.
- A higher priority ring can have its master node located in any position; although, where possible, avoid connecting a common segment to the secondary port of a master node.
- Do not locate the master node on a segment that is shared with a higher priority ring, but you "can" locate it on a common segment that is shared with a lower priority ring. In this situation however, the port that connects to the common segment must be configured as the primary port.

  For example, in Figure 56-12, the upper portion of Node D could be configured as a Master Node of the upper ring (having a priority of 120), but its lower portion must be configured as a Transit Node (having the lower priority of 60).
- On the transit nodes that connect to shared links, allocate the ring's priority to the ports that connect to each ring. Note that both of these nodes "must" be set to the same priority value.

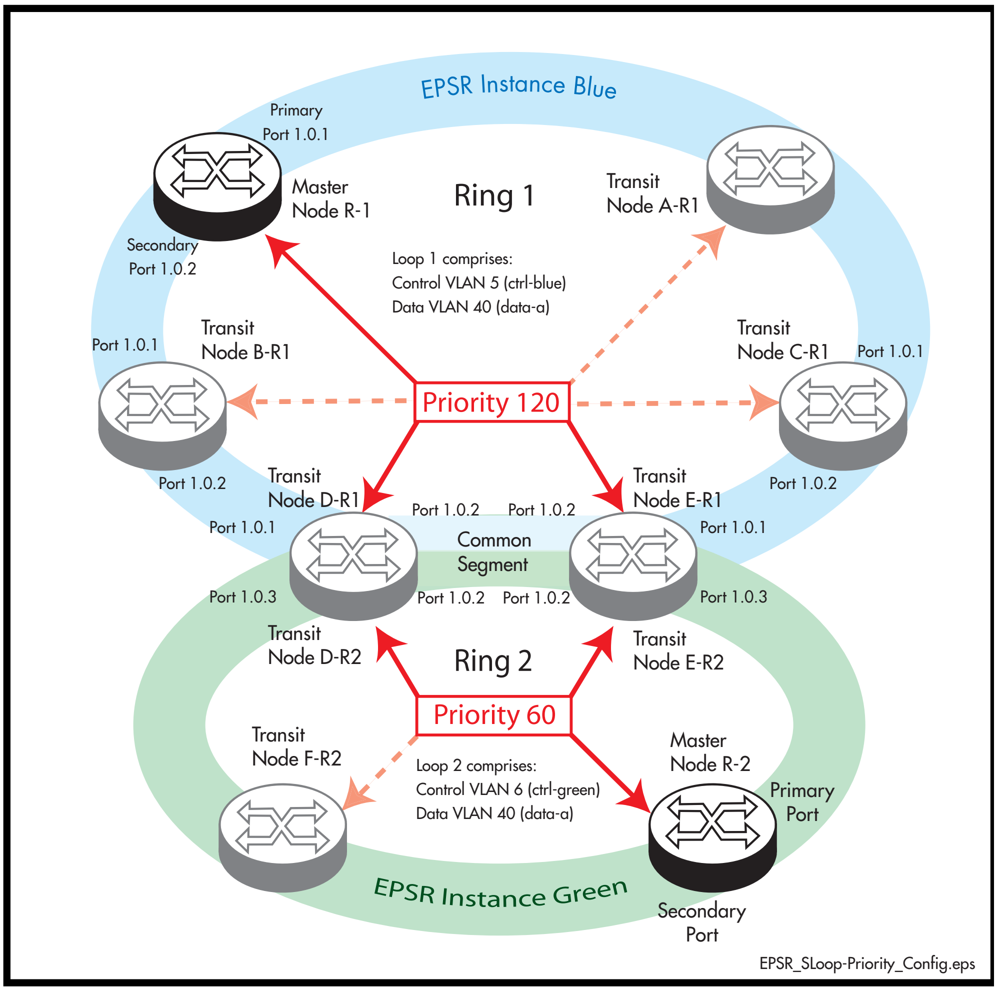
Note For good practice, we advise that you set all nodes within a ring to the priority assigned to that ring. So, for the network of Figure 56-12 each of the nodes that form part of the upper ring would be configured with a priority of 120, and each of the nodes that form the lower ring would all be configured with a priority of 60.

Configuring a Basic Superloop Protected Two Ring EPSR Network

Configuration Example

This section shows how to configure a basic EPSR network such as that shown in Figure 56-13 below

Figure 56-13: EPSR Two Shared Ring Example



The configuration suggested comprises the following basic steps:

- "On Ring 1- Configure the Master Node R-1" on page 56.22
- "On Ring 1 - Configure the Transit Nodes A to C" on page 56.24
- "On Ring 2 - Configure the Master Node R-2" on page 56.26
- "On Rings 1 and 2 - Configure the Transit Nodes D and E" on page 56.28
- "On Ring 2 - Configure the Transit Node F" on page 56.33

On Ring 1- Configure the Master Node R-1

Step 1: Create the control and data VLANs (Configure on the Master Node R-1)

awplus# configure terminal	Enter the Global Configuration mode.
awplus(config)# vlan database	Enter the VLAN Configuration mode.
awplus(config-vlan)# vlan 5 name ctrl-blue state enable	Enable VLAN 5 called ctrl-blue on the Master Node R-1. Specifying the enable state allows forwarding of frames on the VLAN-aware node.
awplus(config-vlan)# vlan 40 name data-a state enable	Enable VLAN 40 called **data-a** on the Master Node R-1. Specifying the enable state allows forwarding of frames on the VLAN-aware node.
awplus(config-vlan)# exit	Exit the VLAN Configuration mode and enter the Global Configuration mode.

Step 2: Add the control VLAN (ctrl-blue) to the Ring Ports

awplus(config)# interface port1.0.1,port1.0.2	Specify the two ring ports (port1.0.1 and port1.0.2) that you are configuring and enter the Interface Configuration mode.
awplus(config-if)# switchport mode trunk	Set the switching characteristics of these ports to Trunk mode.
awplus(config-if)# switchport trunk allowed vlan add 5	Enable VLAN 5 on these ports.
awplus# switchport trunk native vlan none	Remove the native VLAN from these ring ports.

Step 3: Create the EPSR Instance called “blue”, make VLAN 5 the control VLAN and port 1.0.1 the primary port (Configure on the Master Node R-1)

awplus(config)# epsr configuration	Enter the EPSR Configuration mode.
awplus(config-epsr)# epsr blue mode master controlvlan 5 primaryport port1.0.1	Create an EPSR instance called blue on vlan 5. Make vlan 5 the control VLAN. Make port 1.0.1 the primary port. Make this node the master.

Step 4: Add a data VLAN to the EPSR Instance called “blue” (Configure on the Master Node R-1)

awplus(config-epsr)# epsr blue datavlan 40	On epsr instance called blue data-a the data VLAN.

Step 5: Assign a priority to the ring instance (Configure on the Master Node R-1)

awplus(config-epsr)# epsr blue priority 120	Set the ring instance priority to the value selected for the ring. The priority value selected is 120.

Step 6: Enable the EPSR Instance called “blue” (Configure on the Master Node R-1)

awplus(config-epsr)# epsr blue state enable	Enable the EPSR instance named blue.
awplus(config-epsr)# exit	Exit the EPSR Configuration mode.

Step 7: Add port1.0.1 to these VLANs (Configure on the Master Node R-1)

awplus(config)# interface port1.0.1,port1.0.2	Specify the EPSR ring ports (port1.0.1 and 1.0.2) that you are configuring and enter the Interface Configuration mode.
awplus(config-if)# switchport trunk allowed vlan add 40	Enable VLAN 40 on this port.
awplus(config-if)# exit	Exit the Interface mode and enter the Global Configuration mode.

On Ring 1 - Configure the Transit Nodes A to C

Step 1: Create the control and data VLANs (on Transit Nodes A to C)

awplus# `configure terminal`	Enter the Global Configuration mode.
awplus(config)# `vlan database`	Enter the VLAN Configuration mode.
awplus(config-vlan)# `vlan 5 name ctrl-blue state enable`	Enable VLAN 5 called `ctrl-blue` on the Transit Node. Specifying the enable state allows forwarding of frames on the VLAN-aware node.
awplus(config-vlan)# `vlan 40 name data-a state enable`	Enable VLAN 40 called `data-a` on the Transit Node. Specifying the enable state allows forwarding of frames on the VLAN-aware node.
awplus(config-vlan)# `exit`	Exit the VLAN Configuration mode and enter the Global Configuration mode.

Step 2: Add the EPSR control vlan (ctrl-blue) to EPSR ring ports

awplus(config)# `interface port1.0.1,port1.0.2`	Specify the two ring ports (`port1.0.1 and port1.0.2`) that you are configuring and enter the Interface Configuration mode.
awplus(config-if)# `switchport mode trunk`	Set the switching characteristics of this port to Trunk mode.
awplus(config-if)# `switchport trunk allowed vlan add 5`	Enable VLAN 5 on these ports.
awplus(config-if)# `switchport trunk native vlan none`	Remove the native VLAN from the ring ports.

Step 3: Create the EPSR Instance called "blue", make VLAN 5 the control VLAN (on Transit Nodes A to C)

awplus(config)# `epsr configuration`	Enter the EPSR Configuration mode.
awplus(config-epsr)# `epsr blue mode transit controlvlan 5`	Create an EPSR instance called `blue` on `vlan 5`. Make `vlan 5` the control VLAN. Make this node a transit node.

Step 4: Add a data VLAN to the EPSR Instance called “blue” (on Transit Nodes A to C)

awplus(config-epsr)# epsr blue datavlan 40	On the EPSR instance called blue make vlan 40 the data VLAN.

Step 5: Assign a priority to the ring instance (on Transit Nodes A to C)

This step is **mandatory on transit nodes that connect to a common segment**, and good practice on other transit nodes.

awplus(config-epsr)# epsr blue priority 120	Set the ring instance priority to the priority selected for the ring 120.

Step 6: Enable the EPSR Instance called “blue” (on Transit Nodes A to C)

awplus(config-epsr)# epsr blue state enable	Enable the EPSR instance named blue.
awplus(config-epsr)# exit	Exit the EPSR Configuration mode.

Step 7: Add the physical port 1.0.1 to VLAN 40 (on Transit Nodes A to C)

awplus(config)# interface port1.0.1,port1.0.2	Specify the physical ring ports (ports1.0.1 and ports 1.0.2) that you are configuring and enter the Interface Configuration mode.
awplus(config-if)# switchport trunk allowed vlan add 40	Enable VLAN 40 on this port.
awplus(config-if)# exit	Exit the Interface mode and enter the Global Configuration mode.

On Ring 2 - Configure the Master Node R-2

Step 1: Create the control and data VLANs (Configure on the Master Node R-2)

`awplus#`
`configure terminal`
	Enter the Global Configuration mode.

`awplus(config)#`
`vlan database`
	Enter the VLAN Configuration mode.

`awplus(config-vlan)#`
`vlan 6 name ctrl-green state enable`
	Enable vlan 6 called ctrl-green on the Master Node R-2. Specifying the enable state allows forwarding of frames on the VLAN-aware node.

`awplus(config-vlan)#`
`vlan 40 name data-a state enable`
	Enable VLAN 40 called data-a on the Master Node R-2. Specifying the enable state allows forwarding of frames on the VLAN-aware node.

`awplus(config-vlan)#`
`exit`
	Exit the VLAN Configuration mode and enter the Global Configuration mode.

Step 2: Add the control VLAN (ctrl-green) to the Ring Ports

`awplus(config)#`
`interface port1.0.1,port1.0.2`
	Specify the ports (`port1.0.1 and port1.0.2`) that you are configuring, and enter the Interface Configuration mode.

`awplus(config-if)#`
`switchport mode trunk`
	Set the switching characteristics of these ports to Trunk mode.

`awplus(config-if)#`
`switchport trunk allowed vlan add 6`
	Enable vlan 6 on these ports.

`awplus(config-if)#`
`switchport trunk native vlan none`
	Remove the native VLAN from these ring ports.

Step 3: Create the EPSR Instance called “green”, make vlan 6 the control VLAN and port1.0.1 the primary port (Configure on the Master Node R-2)

awplus(config)# `epsr configuration`	Enter the EPSR Configuration mode.
awplus(config-epsr)# `epsr green mode master controlvlan 6 primaryport port1.0.1`	Create an EPSR instance called `ctrl-green` on `vlan 6`. Make `vlan 6` the control VLAN. Make `port 1.0.1` the primary port. Make this node the master.

Step 4: Add a data VLAN to the EPSR Instance called “green” (Configure on the Master Node R-2)

awplus(config-epsr)# `epsr green datavlan 40`	On epsr instance called green make `vlan 40` the data VLAN.

Step 5: Assign a priority to the ring instance (Configure on the Master Node R-2)

This step is **mandatory on transit nodes that connect to a common segment**, and good practice on other transit nodes.

awplus(config-epsr)# `epsr green priority 60`	Set the ring instance priority to the value selected for the ring. The priority value selected is 60.

Step 6: Enable the EPSR Instance called “green” (Configure on the Master Node R-2)

awplus(config-epsr)# `epsr green state enable`	Enable the EPSR instance named green.
awplus(config-epsr)# `exit`	Exit the EPSR Configuration mode.

Step 7: Add ports 1.0.1 and 1.0.2 to these VLANs (Configure on the Master Node R-2)

awplus(config)# interface port1.0.1,port1.0.2	Specify the ports (port1.0.1 and port1.0.2) that you are configuring and enter the Interface Configuration mode.
awplus(config-if)# switchport mode trunk	Set the switching characteristics of these ports to Trunk mode.
awplus(config-if)# switchport trunk allowed vlan add 40	Enable VLAN 40 on this port
awplus(config-if)# exit	Exit the Interface mode and enter the Global Configuration mode.

On Rings 1 and 2 - Configure the Transit Nodes D and E

Step 1: Create the control and data VLANs (on Transit Nodes D and E)

awplus# configure terminal	Enter the Global Configuration mode.
awplus(config)# vlan database	Enter the VLAN Configuration mode.
awplus(config-vlan)# vlan 5 name ctrl-blue state enable	Enable VLAN 5 called ctrl-blue on the Transit Node. Specifying the enable state allows forwarding of frames on the VLAN-aware node.
awplus(config-vlan)# vlan 40 name data-a state enable	Enable VLAN 40 called data-a on the Transit Node. Specifying the enable state allows forwarding of frames on the VLAN-aware node.
awplus(config-vlan)# vlan 6 name ctrl-green state enable	Enable VLAN 6 called ctrl-green on the Transit Node. Specifying the enable state allows forwarding of frames on the VLAN-aware node.
awplus(config-vlan)# exit	Exit the VLAN Configuration mode and enter the Global Configuration mode.

Step 2: Add physical port1.0.1 to these VLANs (on Transit Nodes D and E)

awplus(config)# `interface port1.0.1`	Specify the physical `interface (port1.0.1)` that you are configuring and enter the Interface Configuration mode.
awplus(config-if)# `switchport mode trunk`	Set the switching characteristics of this port to Trunk mode.
awplus(config-if)# `switchport trunk allowed vlan add 5`	Enable VLAN 5 on this port.
awplus(config-if)# `switchport trunk native vlan none`	Remove the native VLAN.
awplus(config-if)# `exit`	Exit the Interface mode and enter the Global Configuration mode.

Step 3: Add physical port port1.0.2 to these VLANs (on Transit Nodes D and E)

awplus(config)# `interface port1.0.2`	Specify the physical `interface (port1.0.2)` that you are configuring and enter the Interface Configuration mode.
awplus(config-if)# `switchport mode trunk`	Set the switching characteristics of this port to Trunk mode.
awplus(config-if)# `switchport trunk allowed vlan add 5`	Enable VLAN 5 (ctrl-blue) on this port.
awplus(config-if)# `switchport trunk allowed vlan add 6`	Enable VLAN 6 (ctrl-green) on this port.
awplus(config-if)# `switchport trunk native vlan none`	Remove the native VLAN.
awplus(config-if)# `exit`	Exit the Interface mode and enter the Global Configuration mode.

Step 4: Add physical port1.0.3 to these VLANs (on Transit Nodes D and E)

awplus(config)# `interface port1.0.3`	Specify the physical `interface (port1.0.3)` that you are configuring and enter the Interface Configuration mode.
awplus(config-if)# `switchport mode trunk`	Set the switching characteristics of this port to Trunk mode.

`awplus(config-if)#`
`switchport trunk allowed vlan add 6`
	Enable VLAN 6 on this port.

`awplus(config-if)#`
`switchport trunk native vlan none`
	Remove the native VLAN.

`awplus(config-if)#`
`exit`
	Exit the Interface mode and enter the Global Configuration mode.

Step 5: Create the EPSR Instance called "blue" on a transit node, make VLAN 5 the control VLAN (on Transit Nodes D and E)

`awplus(config)#`
`epsr configuration`
	Enter the EPSR Configuration mode.

`awplus(config-epsr)#`
`epsr blue mode transit controlvlan 5`
	Create an EPSR instance called `blue` on `vlan 5`.
	Make `vlan 5` the control VLAN.
	Make this node a transit node.

Step 6: Add a data VLAN to the EPSR Instance called "blue" (on Transit Nodes D and E)

`awplus(config-epsr)#`
`epsr blue datavlan 40`
	On the EPSR instance called `blue` make `vlan 40` the data VLAN.

Step 7: Assign a priority to the ring instance (on Transit Nodes D and E)

This step is **mandatory on transit nodes that connect to a common segment**, and good practice on other transit nodes.

`awplus(config-epsr)#`
`epsr blue priority 120`
	Set the ring instance priority to 120 - the value selected for the ring.

`awplus(config-epsr)#`
`exit`
	Exit the EPSR Configuration mode.

Step 8: Enable the EPSR Instance called "blue" (on Transit Nodes D and E)

`awplus(config-epsr)#`
`epsr blue state enable`
	Enable the EPSR instance named `blue`.

Step 9: Create the EPSR Instance called “green” on a transit node, make VLAN 6 the control VLAN (on Transit Nodes D and E)

awplus(config-epsr)# `epsr green mode transit controlvlan 6`	Create an EPSR instance called `green` on `vlan 6`. Make `vlan 6` the control VLAN. Make this node a transit node.

Step 10: Add a data VLAN to the EPSR Instance called “green” (on Transit Nodes D and E)

awplus(config-epsr)# `epsr green datavlan 40`	On the EPSR instance called `green` make `vlan 40` the data VLAN.

Step 11: Assign a priority to the ring instances (on Transit Nodes D and E)

This step is **mandatory on transit nodes that connect to a common segment**, and good practice on other transit nodes.

awplus(config-epsr)# `epsr green priority 60`	Set the ring instance priority to 60 - this being the priority selected for the ring.
awplus(config-epsr)# `exit`	Exit the EPSR Configuration mode.

Step 12: Enable the EPSR Instance called “green” (on Transit Nodes D and E)

awplus(config-epsr)# `epsr green state enable`	Enable the EPSR instance named `green`.
awplus(config-epsr)# `exit`	Exit the EPSR Configuration mode.

Step 13: Add the physical port 1.0.1 to these VLANs (on Transit Nodes D and E)

awplus(config)# `interface port1.0.1`	Specify the physical interface (`port1.0.1`) that you are configuring, and enter the Interface Configuration mode.
awplus(config-if)# `switchport mode trunk`	Set the switching characteristics of this port to Trunk mode.
awplus(config-if)# `switchport trunk allowed vlan add 40`	Enable VLAN 40 on this port.
awplus(config-if)# `exit`	Exit the Interface Configuration mode and enter the Global Configuration mode.

Step 14: Add the physical port1.0.2to these VLANs (on Transit Nodes D and E)

`awplus(config)#`
`interface port1.0.2`
	Specify the physical interface (`port1.0.2`) that you are configuring and enter the Interface Configuration mode.

`awplus(config-if)#`
`switchport mode trunk`
	Set the switching characteristics of this port to Trunk mode.

`awplus(config-if)#`
`switchport trunk allowed vlan add 40`
	Enable VLAN 40 on this port.

`awplus(config-if)#`
`exit`
	Exit the Interface Configuration mode and enter the Global Configuration mode.

Step 15: Add the physical port1.03to these VLANs (on Transit Nodes D and E)

`awplus(config)#`
`interface port1.0.3`
	Specify the physical interface (`port1.0.3`) that you are configuring and enter the Interface Configuration mode.

`awplus(config-if)#`
`switchport mode trunk`
	Set the switching characteristics of this port to Trunk mode.

`awplus(config-if)#`
`switchport trunk allowed vlan add 40`
	Enable VLAN 40 on this port.

`awplus(config-if)#`
`exit`
	Exit the Interface Configuration mode and enter the Global Configuration mode.

On Ring 2 - Configure the Transit Node F

Step 1: Create the control and data VLANs (on Transit Node F)

`awplus#`
`configure terminal`
	Enter the Global Configuration mode.

`awplus(config)#`
`vlan database`
	Enter the VLAN Configuration mode.

`awplus(config-vlan)#`
`vlan 6 name ctrl-green state enable`
	Enable VLAN 6 called `ctrl-green` on the Transit Node. Specifying the enable state allows forwarding of frames on the VLAN-aware node.

`awplus(config-vlan)#`
`vlan 40 name data-a state enable`
	Enable VLAN 40 called `data-a` on the Transit Node. Specifying the enable state allows forwarding of frames on the VLAN-aware node.

`awplus(config-vlan)#`
`exit`
	Exit the VLAN Configuration mode and enter the Global Configuration mode.

Step 2: Create the EPSR Instance called "green" on a transit node, make VLAN 6 the control VLAN (on Transit Node F)

`awplus(config)#`
`epsr configuration`
	Enter the EPSR Configuration mode.

`awplus(config-epsr)#`
`epsr green mode transit controlvlan 6`
	Create an EPSR instance called `green` on `vlan 6`.
	Make `vlan 6` the control VLAN.
	Make this node a transit node.

Step 3: Add a data VLAN to the EPSR Instance called "green" (on Transit Node F)

`awplus(config-epsr)#`
`epsr green datavlan 40`
	On the EPSR instance called `green` make `vlan 40` the data VLAN.

Step 4: Enable the EPSR Instance called "green" (on Transit Node F)

`awplus(config-epsr)#`
`epsr green state enable`
	Enable the EPSR instance named `green`.

Step 5: Assign a priority to the ring instance (on Transit Node F)

This step is **mandatory on transit nodes that connect to a common segment**, and good practice on other transit nodes.

awplus(config-epsr)# epsr green priority 120	Set the ring instance priority to the priority selected for the ring 120.
awplus(config-epsr)# exit	Exit the EPSR Configuration mode.

Step 6: Add the physical port port1.0.1 to VLANs 6 and 40 (on Transit Node F)

awplus(config)# interface port1.0.1	Specify the physical interface (`port1.0.1`) that you are configuring and enter the Interface Configuration mode.
awplus(config-if)# switchport mode trunk	Set the switching characteristics of this port to Trunk mode.
awplus(config-if)# switchport trunk allowed vlan add 6	Enable VLAN 6 on this port.
awplus(config-if)# switchport trunk allowed vlan add 40	Enable VLAN 40 on this port.
awplus(config-if)# switchport trunk native vlan none	Remove the native VLAN
awplus(config-if)# exit	Exit the Interface mode and enter the Global Configuration mode.

Step 7: Add the physical port port1.0.2 to VLANs 6 and 40 (on Transit Node F)

`awplus(config)#`
`interface port1.0.2`
	Specify the interface (`port1.0.2`) that you are configuring and enter the Interface Configuration mode.

`awplus(config-if)#`
`switchport mode trunk`
	Set the switching characteristics of this port to Trunk mode.

`awplus(config-if)#`
`switchport trunk allowed vlan add 6`
	Enable VLAN 6 on this port.

`awplus(config-if)#`
`switchport trunk allowed vlan add 40`
	Enable VLAN 40 on this port.

`awplus(config-if)#`
`switchport trunk native vlan none`
	Remove the native VLAN

`awplus(config-if)#`
`exit`
	Exit the Interface Configuration mode and enter the Global Configuration mode.

Sample Show Output

For the above network configuration, running the command show epsr on node R1 will display the following output when operating normally. Note the blocked state of its secondary port.

Figure 56-14: Output from the **show epsr** command run on Master Node R1 - with Ring 1 - EPSR Instance blue operating normally

```
 EPSR Information
----------------------------------------------------------------
 Name ......................blue
 Mode ........................Master
 Status ......................Enabled
 State .......................Complete
 Control Vlan ................5
 Data VLAN(s) ................40
 Interface Mode ..............Ports Only
 Primary Port ................port1.0.1
   Status ....................Forwarding
   Is On Common Segment ......No
   Blocking Control ..........Physical
 Secondary Port ..............port1.0.2
   Status ....................Blocked
   Is On Common Segment ......No
   Blocking Control ..........Physical
 Hello Time ..................1 s
 Failover Time ...............2 s
 Ring Flap Time ..............0 s
 Trap ........................Enabled
 Enhanced Recovery ...........Disabled
 Priority ....................120
----------------------------------------------------------------
```

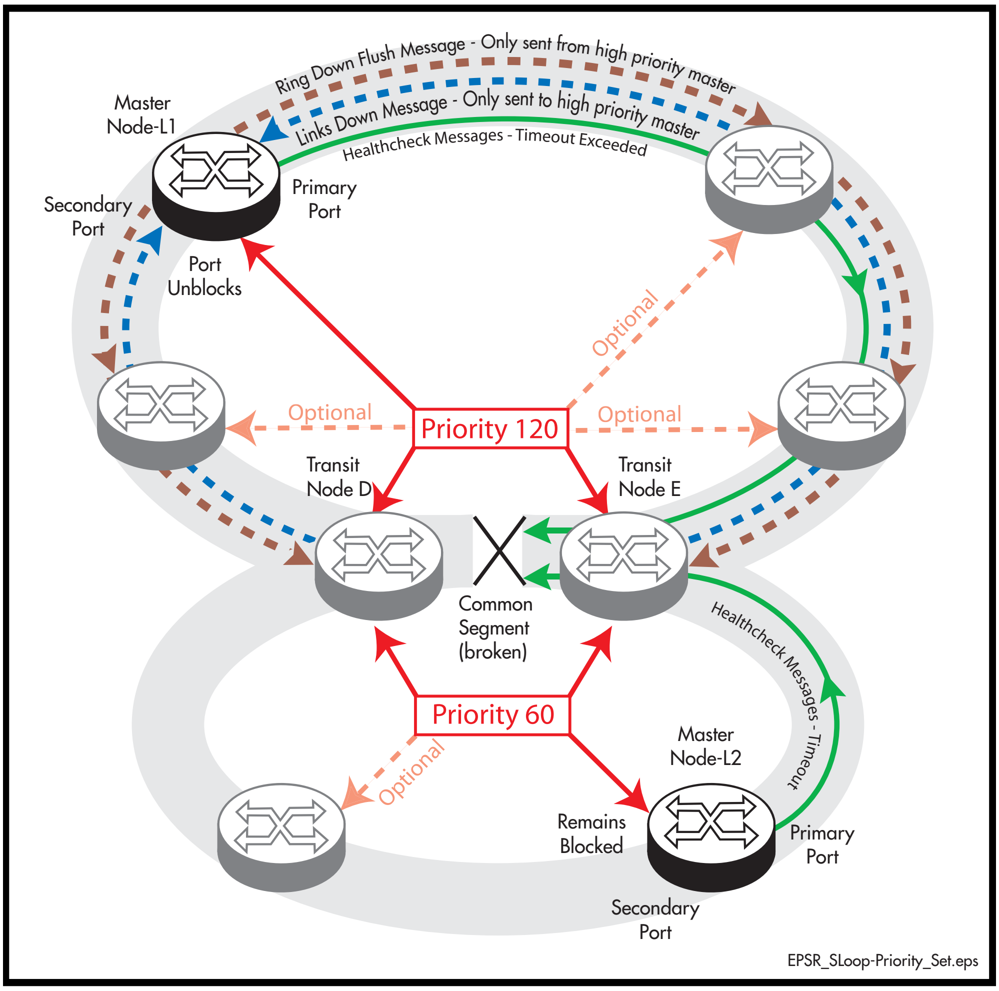
If a fault occurs somewhere within the blue network ring the Master Node-R1 would respond by placing its secondary port into the forwarding state. Figure Figure 56-15 displays its resultant state. Note that the state of its secondary port has now moved from Blocked, Forwarding.

Figure 56-15: Output from the **show epsr** command run on Master Node R2, where a break exists within the Ring 1 - EPSR instance blue.

```
 EPSR Information
----------------------------------------------------------------
 Name ......................blue
 Mode ........................Master
 Status ......................Enabled
 State .......................Failed
 Control Vlan ................6
 Data VLAN(s) ................40
 Interface Mode ..............Ports Only
 Primary Port ................port1.0.1
   Status ....................Forwarding
   Is On Common Segment ......No
   Blocking Control ..........Physical
 Secondary Port ..............port1.0.2
   Status ....................Forwarding
   Is On Common Segment ......No
   Blocking Control ..........Physical
 Hello Time ..................1 s
 Failover Time ...............2 s
 Ring Flap Time ..............0 s
 Trap ........................Enabled
 Enhanced Recovery ...........Disabled
 Priority ....................60
----------------------------------------------------------------
```

If a fault occurs in the common segment of the ring then the Master Node-R2 being on the lower priority ring would detect a timeout of its transmitted Healthcheck Message. It would also detect the absence of the expected **Ring Down Flush** message, see Figure 56-16. The Master node then assumes that there is a break somewhere in the Common Segment, and will display the status shown in Figure 56-17.

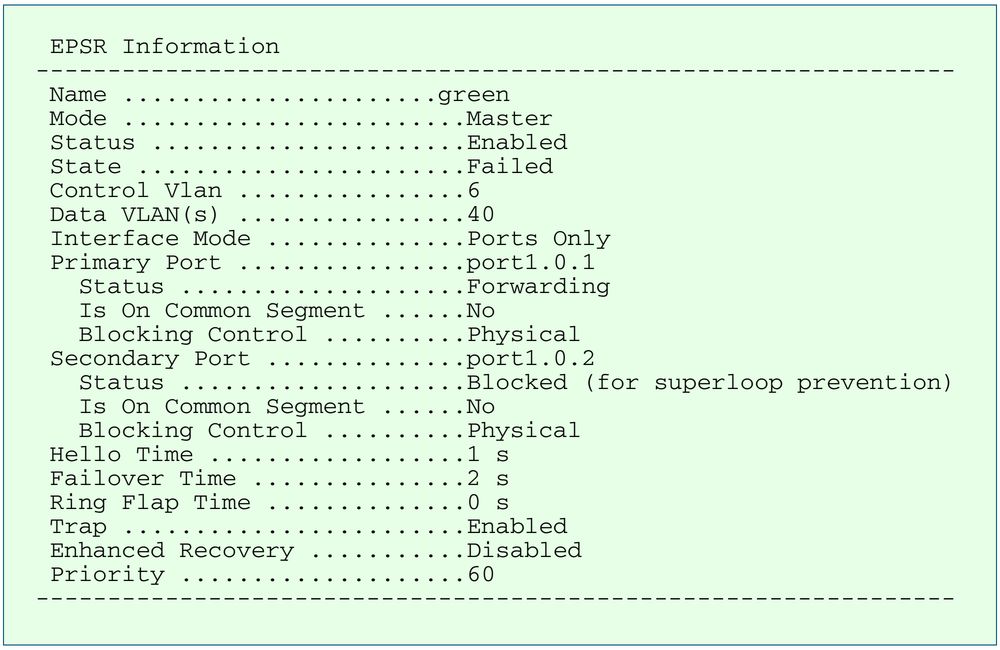
Figure 56-16: EPSR behavior with a faulty common segment and Superloop enabled



Note that the secondary port on Master Node-L2 remains in the blocked state; its state now appears in show output as being as blocked (for superloop prevention), See Figure 56-17.

The Master-L1 on the blue ring will also detect a timeout in the healthcheck message, but because ring 1 has the higher priority (of 120), it will receive a Links Down message from each of the Transit Nodes (D and E) that connect to the common segment. As a result, the state of the Master Node will be as shown in Figure Figure 56-17; note particularly the change in its Secondary Port status.

Figure 56-17: Output from the **show epsr** command run on Master Node L2 (green)



```
 EPSR Information
----------------------------------------------------------------
 Name ......................green
 Mode ........................Master
 Status ......................Enabled
 State .......................Failed
 Control Vlan ................6
 Data VLAN(s) ................40
 Interface Mode ..............Ports Only
 Primary Port ................port1.0.1
   Status ....................Forwarding
   Is On Common Segment ......No
   Blocking Control ..........Physical
 Secondary Port ..............port1.0.2
   Status ....................Blocked (for superloop prevention)
   Is On Common Segment ......No
   Blocking Control ..........Physical
 Hello Time ..................1 s
 Failover Time ...............2 s
 Ring Flap Time ..............0 s
 Trap ........................Enabled
 Enhanced Recovery ...........Disabled
 Priority ....................60
----------------------------------------------------------------
```

Adding a new data VLAN to a functioning superloop topology

This example shows how to add another data VLAN called **data-b** to the superloop topology. We recommend that you apply the configuration steps in the order shown.

1. Add VLAN to the common segment (for both instances)
2. Add VLAN to blue master
3. Add VLAN to other blue transits
4. Add VLAN to green master
5. Add VLAN to other green transits

On Ring 1 EPSR Instance Blue - Configure each of the Transit Nodes that Connect to the Common Segment

Select one of the transit nodes that connects to the common segment, and carry out the following steps:

Step 1: Add VLAN 50 to the VLAN database and set its state to enable

`awplus#`
`configure terminal`
	Enter terminal config mode

`awplus(config)#`
`vlan database`
	Enter the EPSR Configuration mode.

`awplus(config-epsr)#`
`vlan 50 name data-b enable`
	Create vlan 50, name it data-b and enable it.

Step 2: Add the VLAN 50 to the EPSR Instances called “blue” and “green” on the transit nodes

`awplus(config)#`
`epsr configuration`
	Enter the EPSR Configuration mode.

`awplus(config-epsr)#`
`epsr blue datavlan 50`
	On the EPSR instance called `blue` add `vlan 50` as a data VLAN.

`awplus(config-epsr)#`
`epsr green datavlan 50`
	On the EPSR instance called `green` add `vlan 50` as a data VLAN.

Step 3: Add the common physical port (port1.0.2 in this example) to VLAN 50

`awplus(config)#`
`interface port1.0.2`
	Specify the physical interface (`port1.0.2`) that you are configuring and enter the Interface Configuration mode.

awplus(config-if)# switchport trunk allowed vlan add 50	Enable VLAN 50 on this port.
awplus(config-if)# exit	Exit the Interface mode and enter the Global Configuration mode.

Step 4: Add physical port1.0.1 to VLAN 50

awplus(config)# interface port1.0.1	Specify the interface (port1.0.1) that you are configuring and enter the Interface Configuration mode.
awplus(config-if)# switchport trunk allowed vlan add 50	Enable VLAN 50 on this port.
awplus(config-if)# exit	Exit the Interface Configuration mode and enter the Global Configuration mode.

Step 5: Add physical port1.0.3 to VLAN 50

awplus(config)# interface port1.0.3	Specify the interface (port1.0.3) that you are configuring and enter the Interface Configuration mode.
awplus(config-if)# switchport trunk allowed vlan add 50	Enable VLAN 50 on this port.
awplus(config-if)# exit	Exit the Interface Configuration mode and enter the Global Configuration mode.

Select the next transit node that connects to the common segment, and repeat the above steps:

On Ring 1 EPSR Instance Blue - Add VLAN 50 to the Master Node

Carry out this process using the same basic procedure shown in of Steps 1 to 5

On Ring 1 EPSR Instance Blue - Add VLAN 50 to the Transit Nodes

Carry out this process using the same basic procedure shown in of Steps 1 to 5

On Ring 2 EPSR Instance Green - Add VLAN 50 to the Master Node

Carry out this process using the same basic procedure shown in of Steps 1 to 5

On Ring 2 EPSR Instance Green - Add VLAN 50 to the remaining Transit Node

Carry out this process using the same basic procedure shown in of Steps 1 to 5

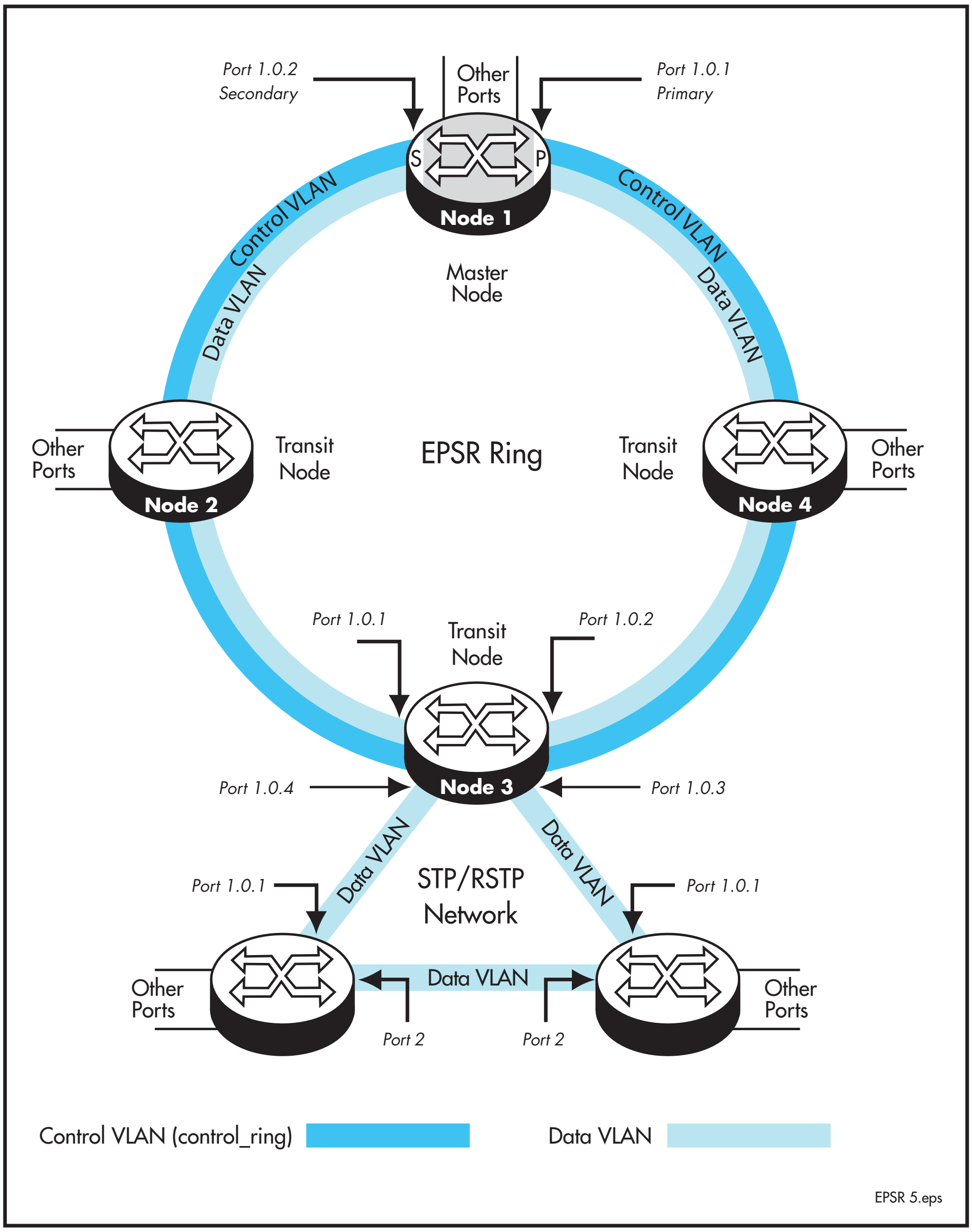
EPSR and Spanning Tree Operation

EPSR and the Spanning Tree protocol (STP) address data loop prevention, although they do it differently. EPSR is manually configured to explicitly identify which links are broken in the defined ring, whereas STP/RSTP calculates where to break links based on user-provided values (metrics) that are compared to determine the "best" (or lowest cost) paths for data traffic.

At the practical level you can use these two techniques to create complementary hybrid EPSR /STP configurations. This configuration might have a high speed fibre loop topology backbone-controlled and managed using EPSR. Lobes could extend out from each loop node into a user mesh network. Any loops in this mesh network would be controlled and managed using STP/RSTP. Note that EPSR and STP cannot share the same ports.

The following figure shows a basic combined EPSR / STP network.

Figure 56-18: EPSR and spanning tree operation



Chapter 57: EPSR Commands

Command List

This chapter provides an alphabetical reference for commands used to configure EPSR. For more information, see Chapter 56, EPSR Introduction and Configuration.

For information about modifying or redirecting the output from **show** commands to a file, see

"Controlling "show" Command Output" on page 1.41.

debug epsr

This command enables EPSR debugging.

The **no** variant of this command disables EPSR debugging.

Syntax

```
debug epsr {info|msg|pkt|state|timer|all}

no debug epsr {info|msg|pkt|state|timer|all}
```

Parameter	Description
`info`	Send general EPSR information to the console. Using this parameter with the **no debug epsr** command will explicitly exclude the above information from being sent to the console.
`msg`	Send the decoded received and transmitted EPSR packets to the console. Using this parameter with the **no debug epsr** command will explicitly exclude the above packets from being sent to the console.
`pkt`	Send the received and transmitted EPSR packets as raw ASCII text to the console. Using this parameter with the **no debug epsr** command will explicitly exclude the above packets from being sent to the console.
`state`	Send EPSR state transitions to the console. Using this parameter with the **no debug epsr** command will explicitly exclude state transitions from being sent to the console.
`timer`	Send EPSR timer information to the console. Using this parameter with the **no debug epsr** command will explicitly exclude timer information from being sent to the console.
`all`	Send all EPSR debugging information to the console. Using this parameter with the **no debug epsr** command will explicitly exclude any debugging information from being sent to the console.

Mode Privileged Exec

Examples To enable state transition debugging, use the command:

```
awplus# debug epsr state
```

To disable EPSR packet debugging, use the command:

```
awplus# no debug epsr pkt
```

Related Commands undebug epsr

epsr

This command sets the timer values for an EPSR instance. It is only valid for master nodes.

The **no** variant of this command destroys an EPSR instance.

Syntax

```
epsr <epsr-name> {hellotime <1-32767>|failovertime <2-65535>|
    ringflaptime <0-65535>}
```

```
no epsr <epsr-name>
```

Caution Using the "no" variant of this command will remove the specified epsr instance.

Parameter	Description
`<epsr-name>`	Name of the EPSR instance.
`hellotime <1-32767>`	The number of seconds between the transmission of health check messages.
`failovertime <2-65535>`	The number of seconds that a master waits for a returning health check message before entering the failed state. **The failover time must be at least twice the hellotime.** This is to force the master node to wait until it detects the absence of two sequential healthcheck messages before it enters the failed state.
`ringflaptime <0-65535>`	The minimum number of seconds that a master must remain in the failed state.

Mode EPSR Configuration

Examples To set the hellotimer to 5 seconds for the EPSR instance called `blue`, use the command:

```
awplus(config-epsr)# epsr blue hellotime 5
```

Note When VCStack is used with EPSR, the EPSR failovertime should be at least 5 seconds. see the epsr command on page 57.4.

To destroy an EPSR instance called `blue`, use the command:

```
awplus(config-epsr)# no epsr blue
```

Related Commands

epsr mode master controlvlan primaryport
epsr mode transit controlvlan
epsr configuration
epsr datavlan
epsr state
epsr trap
reboot rolling
show epsr

epsr configuration

Use this command to enter EPSR Configuration mode so that EPSR can be configured.

Syntax

```
epsr configuration
```

Mode Global Configuration

Example To change to EPSR mode, use the command:

```
awplus(config)# epsr configuration
```

Related Commands

epsr mode master controlvlan primaryport
epsr
show epsr

epsr datavlan

This command adds a data VLAN or a range of VLAN identifiers to a specified EPSR instance.

The **no** variant of this command removes a data vlan or data vlan range from an EPSR instance.

Syntax

```
epsr <epsr-name> datavlan {<vlanid>|<vlanid-range>}

no epsr <epsr-name> datavlan {<vlanid>|<vlanid-range>}
```

Parameter	Description
<epsr-name>	Name of the EPSR instance.
`datavlan`	Adds a data VLAN to be protected by the EPSR instance.
<vlanid>	The VLAN's VID - a number between 1 and 4094 excluding the number selected for the control VLAN.
<vlanid-range>	Specify a range of VLAN identifiers using a hyphen to separate identifiers.

Mode EPSR Configuration

Usage We suggest setting the epsr controlvlan to vlan2 using the epsr mode master controlvlan primaryport and epsr mode transit controlvlan commands, then setting the EPSR data VLAN between to be a value 3 and 4094 using the epsr datavlan command.

Examples To add `vlan3` to the EPSR instance called `blue`, use the command:

```
awplus(config-epsr)# epsr blue datavlan vlan3
```

To add `vlan2` and `vlan3` to the EPSR instance called `blue`, use the command:

```
awplus(config-epsr)# epsr blue datavlan vlan2-vlan3
```

To remove `vlan3` from the EPSR instance called `blue`, use the command:

```
awplus(config-epsr)# no epsr blue datavlan vlan3
```

To remove `vlan2` and `vlan3` from the EPSR instance called `blue`, use the command:

```
awplus(config-epsr)# no epsr blue datavlan vlan2-vlan3
```

Related Commands

epsr mode master controlvlan primaryport
epsr mode transit controlvlan
show epsr

epsr enhancedrecovery enable

This command enables EPSR's enhanced recovery mode. Enhanced recovery mode enables a ring to apply additional recovery procedures when a ring with more than one break, partially mends. For more information see, "Managing Rings with Two Breaks" on page 56.7.

The **no** variant of this command disables the enhancedrecovery mode.

Syntax

```
epsr <epsr-name> enhancedrecovery enable

no epsr <epsr-name> enhancedrecovery enable
```

Parameter	Description
<epsr-name>	Name of the EPSR instance.

Default Default is enhancedrecovery mode disabled.

Mode EPSR Configuration

Example To apply enhanced recovery on the EPSR instance called `blue`, use the command:

```
awplus(config-epsr)# epsr blue enhancedrecovery enable
```

Related Commands show epsr

epsr mode master controlvlan primaryport

This command creates a master EPSR instance.

Syntax

```
epsr <epsr-name> mode master controlvlan <2-4094> primaryport <port>
```

Parameter	Description
<epsr-name>	Name of the EPSR instance.
`mode`	Determines the node is acting as a master.
`master`	Sets switch to be the master node for the named EPSR ring.
`controlvlan`	The VLAN that will transmit EPSR control frames.
<2-4094>	VLAN ID.
`primaryport`	Primary port for the EPSR instance.
<port>	The primary port. The port may be a switch port (e.g. `port1.0.4`) or a static channel group (e.g. `sa3`). It cannot be a dynamic (LACP) channel group.

Note The software allows you to configure more than two ports or static channel groups to the control VLAN within a single switchor stacked node. However, we advise against this because in certain situations it can produce unpredictable results.

If the control VLAN contains more than two ports (or static channels) an algorithm selects the two ports or channels with the lowest number to be the ring ports. However if the switch has only one channel group is defined to the control vlan, EPSR will not operate on the secondary port.

EPSR does not support Dynamic link aggregation (LACP).

Mode EPSR Configuration

Example To create a master EPSR instance called `blue` with `vlan2` as the control VLAN and `port1.0.1` as the primary port, use the command:

```
awplus(config-epsr)# epsr blue mode master controlvlan vlan2
                     primaryport port1.0.1
```

Related Commands epsr mode transit controlvlan
show epsr

epsr mode transit controlvlan

This command creates a transit EPSR instance.

Syntax

```
epsr <epsr-name> mode transit controlvlan <2-4094>
```

Parameter	Description
`<epsr-name>`	Name of the EPSR instance.
`mode`	Determines the node is acting as a transit node.
`transit`	Sets switch to be the transit node for the named EPSR ring.
`controlvlan`	The VLAN that will transmit EPSR control.
`<2-4094>`	VLAN id.

Note The software allows you to configure more than two ports or static channel groups to the control VLAN within a single switch or stacked node. However, we advise against this because in certain situations it can produce unpredictable results.

If the control VLAN contains more than two ports (or static channels) an algorithm selects the two ports or channels with the lowest number to be the ring ports. However if the switch has only one channel group is defined to the control vlan, EPSR will not operate on the secondary port.

EPSR does not support Dynamic link aggregation (LACP).

Mode EPSR Configuration

Example To create a transit EPSR instance called `blue` with `vlan2` as the control VLAN, use the command:

```
awplus(config-epsr)# epsr blue mode transit controlvlan vlan2
```

Related Commands

epsr mode master controlvlan primaryport
epsr mode transit controlvlan
show epsr

epsr priority

This command sets the priority of an EPSR instance on an EPSR node. Priority is used to prevent superloops forming under fault conditions with particular ring configurations. Setting a node to a value greater than one, also has the effect of turning on **superloop protection**.

The **no** variant of this command returns the priority of the EPSR instance back to its default value of 0, which also disables EPSR Superloop prevention.

Syntax

```
epsr <epsr-name> priority <0-127>
```

```
no <epsr-name> priority
```

Parameter	Description
<epsr-name>	Name of the EPSR instance.
`priority`	The priority of the ring instance selected by the epsr-name parameter.
<0-127>	The priority to be applied (0 is the lowest priority and represents no superloop protection.

Default The default priority of an EPSR instance on an EPSR node is 0. The negated form of this command resets the priority of an EPSR instance on an EPSR node to the default value.

Mode EPSR Configuration

Example To set the priority of the EPSR instance called `blue` to the highest priority (127), use the command:

```
awplus(config-epsr)# epsr blue priority 127
```

To reset the priority of the EPSR instance called `blue` to the default (0), use the command:

```
awplus(config-epsr)# no epsr blue priority
```

Related Commands epsr configuration

epsr state

This command enables or disables an EPSR instance.

Syntax

```
epsr <epsr-name> state {enabled|disabled}
```

Parameter	Description
`<epsr-name>`	The name of the EPSR instance.
`state`	The operational state of the ring.
`enabled`	EPSR instance is enabled.
`disabled`	EPSR instance is disabled.

Mode EPSR Configuration

Example To enable the EPSR instance called `blue`, use the command:

```
awplus(config-epsr)# epsr blue state enabled
```

Related Commands epsr mode master controlvlan primaryport
epsr mode transit controlvlan

epsr trap

This command enables SNMP traps for an EPSR instance. The traps will be sent when the EPSR instance changes state.

The **no** variant of this command disables SNMP traps for an EPSR instance. The traps will no longer be sent when the EPSR instance changes state.

Syntax

```
epsr <epsr-name> trap
```

```
no epsr <epsr-name> trap
```

Parameter	Description
`<epsr-name>`	Name of the EPSR instance.
`trap`	SNMP trap for the EPSR instance.

Mode EPSR Configuration

Example To enable traps for the EPSR instance called `blue`, use the command:

```
awplus(config-epsr)# epsr blue trap
```

To disable traps for the EPSR instance called `blue`, use the command:

```
awplus(config-epsr)# no epsr blue trap
```

Related Commands epsr mode master controlvlan primaryport
epsr mode transit controlvlan
show epsr

show debugging epsr

This command shows the debugging modes enabled for EPSR.

Syntax

```
show debugging epsr
```

Mode User Exec and Privileged Exec

Example To show the enabled debugging modes, use the command:

```
awplus# show debugging epsr
```

Related Commands debug epsr

show epsr

This command displays information about all EPSR instances.

Syntax `show epsr`

Mode User Exec and Privileged Exec

Example To show the current settings of all EPSR instances, use the command:

```
awplus# show epsr
```

Output The following examples show the output display for a **non** superloop topology network.

Figure 57-1: Example output from the **show epsr** command run on a Master Node

```
EPSR Information
---------------------------------------------------------------
 Name ........................ test2
 Mode .......................... Transit
 Status ........................ Enabled
 State ......................... Links-Up
 Control Vlan .................. 2
 Data VLAN(s) .................. 10
 Interface Mode ................ Ports Only
 First Port .................... port1.0.1
 First Port Status ............. Down
 First Port Direction .......... Unknown
 Second Port ................... port1.0.2
 Second Port Status ............ Down
 Second Port Direction ......... Unknown
 Trap .......................... Enabled
 Master Node ................... Unknown
 Enhanced Recovery ............. Disabled
---------------------------------------------------------------
```

Figure 57-2: Example output from the **show epsr** command run on a Transit Node

```
EPSR Information
---------------------------------------------------------------
 Name ........................ test4
 Mode .......................... Master
 Status ........................ Enabled
 State ......................... Complete
 Control Vlan .................. 4
 Data VLAN(s) .................. 20
 Interface Mode ................ Ports Only
 Primary Port .................. port1.0.3
 Primary Port Status ........... Forwarding
 Secondary Port ................ port1.0.4
 Secondary Port Status ......... Forwarding
 Hello Time .................... 1 s
 Failover Time ................. 2 s
 Ring Flap Time ................ 0 s
 Trap .......................... Enabled
 Enhanced Recovery ............. Disabled
---------------------------------------------------------------
```

The following examples show the output display for a superloop topology network.

The following examples show the output display for superloop topology network.

Figure 57-3: Example output from the **show lacp** command run on a Master Node

```
 EPSR Information
---------------------------------------------------------------
 Name ........................ test4
 Mode .......................... Master
 Status ........................ Enabled
 State ......................... Complete
 Control Vlan .................. 4
 Data VLAN(s) .................. 20
 Interface Mode ................ Ports Only
 Primary Port .................. port1.0.3
   Status ...................... Forwarding (logically blocking)
   Is On Common Segment ........ No
   Blocking Control ............ Physical
 Secondary Port ................ port1.0.4
   Status ...................... Blocked
   Is On Common Segment ........ No
   Blocking Control ............ Physical
 Hello Time .................... 1 s
 Failover Time ................. 2 s
 Ring Flap Time ................ 0 s
 Trap .......................... Enabled
 Enhanced Recovery ............. Disabled
 SLP Priority .................. 12
---------------------------------------------------------------
```

Table 57-1: Parameters displayed in the output of the **show epsr** command

Parameter on Master Node	Parameter on Transit Node	Description
`Name`	`Name`	The name of the EPSR instance.
`Mode`	`Mode`	The mode in which the EPSR instance is configured - either Master or Transit
`Status`	`Status`	Indicates whether the EPSR instance is enabled or disabled
`State`	`State`	Indicates state of the EPSR instance's state machine. Master states are: Idle, Complete, and Failed. Transit states are Links-Up, Links-Down, and Pre-Forwarding.
`Control Vlan`	`Control Vlan`	Displays the VID of the EPSR instance's control VLAN.
`Data VLAN(s)`	`Data VLAN(s)`	The VID(s) of the instance's data VLANs.
`Interface Mode`	`Interface Mode`	Whether the EPSR instance's ring ports are both physical ports (Ports Only) or are both static aggregators (Channel Groups Only).
`Primary Port`	`First Port`	The EPSR instance's primary ring port.
`- Status`	`- Status`	Whether the ring port is forwarding (Forwarding) or blocking (Blocked), or has link down (Down), and if forwarding or blocking, "(logical)" indicates the instance has only logically set the blocking state of the port because it does not have physical control of it.
	`- Direction`	The ring port on which the last EPSR control packet was received is indicated by "Upstream". The other ring port is then "Downstream"

Table 57-1: Parameters displayed in the output of the **show epsr** command

Parameter on Master	Parameter on Transit	Description(cont.)
`- Is On Common Segment`	`- Is On Common Segment`	Whether the ring port is on a shared common segment link to another node, and if so, "(highest rank)" indicates it is the highest priority instance on that common segment.
`- Blocking Control`	`- Blocking Control`	Whether the instance has "physical" or "logical" control of the ring port's blocking in the instance's data VLANs.
`Secondary Port`	`Second Port`	The EPSR instance's secondary port.
`- Status`	`- Status`	Whether the ring port is forwarding (Forwarding) or blocking (Blocked), or has link down (Down), and if forwarding or blocking, "(logical)" indicates the instance has only logically set the blocking state of the port, because it does not have physical control of it. Note that on a master configured for SuperLoop Prevention (non-zero priority) its secondary ring port can be physically forwarding, but logically blocking. This situation arises when it is not the highest priority node in the topology (and so does not receive LINKS-DOWN messages upon common segment breaks) and a break on a common segment in its ring is preventing reception of its own health messages.
	`- Direction`	The ring port on which the last EPSR control packet was received is indicated by "Upstream". The other ring port is then "Downstream"
`- Is On Common Segment`	`- Is On Common Segment`	Whether the ring port is on a shared common segment link to another node, and if so, "(highest rank)" indicates it is the highest priority instance on that common segment
`- Blocking Control`	`- Blocking Control`	Whether the instance has "physical" or "logical" control of the ring port's blocking in the instance's data VLANs
`Hello Time`		The EPSR instance's setting for the interval between transmissions of health check messages (in seconds)
`Failover Time`		The time (in seconds) the EPSR instance waits to receive a health check message before it decides the ring is down
`Ring Flap Time`		The minimum time the EPSR instance must remain in the failed state
`Trap`	`Trap`	Whether the EPSR instance has EPSR SNMP traps enabled
`Enhanced Recovery`	`Enhanced Recovery`	Whether the EPSR instance has enhanced recovery mode enabled
`SLP Priority`	`SLP Priority`	The EPSR instance's priority (for SuperLoop Prevention)

Figure 57-4: Example output from the **show lacp** command run on a Transit Node

```
EPSR Information
---------------------------------------------------------------
 Name ........................ test2
 Mode .......................... Transit
 Status ........................ Enabled
 State ......................... Links-Up
 Control Vlan .................. 2
 Data VLAN(s) .................. 10
 Interface Mode ................ Ports Only
 First Port .................... port1.0.1
   Status ...................... Forwarding
   Direction ................... Downstream
   Is On Common Segment ........ Yes (highest rank)
   Blocking Control ............ Physical
 Second Port ................... port1.0.2
   Status ...................... Forwarding
   Direction ................... Upstream
   Is On Common Segment ........ No
   Blocking Control ............ Physical
 Trap .......................... Enabled
 Master Node ................... Unknown
 Enhanced Recovery ............. Disabled
 SLP Priority .................. 10

 Name ........................ test3
 Mode .......................... Transit
 Status ........................ Enabled
 State ......................... Links-Up
 Control Vlan .................. 3
 Data VLAN(s) .................. 10
 Interface Mode ................ Ports Only
 First Port .................... port1.0.1
   Status ...................... Forwarding (logical)
   Direction ................... Downstream
   Is On Common Segment ........ Yes
   Blocking Control ............ Logical
 Second Port ................... port1.0.3
   Status ...................... Forwarding
   Direction ................... Upstream
   Is On Common Segment ........ No
   Blocking Control ............ Physical
 Trap .......................... Enabled
 Master Node ................... Unknown
 Enhanced Recovery ............. Disabled
 SLP Priority .................. 9
---------------------------------------------------------------
```

Related Commands

epsr mode master controlvlan primaryport
epsr mode transit controlvlan
show epsr counters

show epsr word

This command displays information about the specified EPSR instance.

Syntax

```
show epsr <epsr-name>
```

Parameter	Description
<epsr-name>	Name of the EPSR instance.

Mode User Exec and Privileged Exec

Example To show the current settings of the EPSR instance called `blue`, use the command:

```
awplus# show epsr blue
```

Related Commands

epsr mode master controlvlan primaryport
epsr mode transit controlvlan
show epsr counters

show epsr word counters

This command displays counter information about the specified EPSR instance.

Syntax

```
show epsr <epsr-name> counters
```

Parameter	Description
<epsr-name>	Name of the EPSR instance.

Mode User Exec and Privileged Exec

Example To show the counters of the EPSR instance called `blue`, use the command:

```
awplus# show epsr blue counters
```

Related Commands

epsr mode master controlvlan primaryport
epsr mode transit controlvlan
show epsr

show epsr counters

This command displays counter information about all EPSR instances.

Syntax

```
show epsr counters
```

Mode User Exec and Privileged Exec

Example To show the counters of all EPSR instances, use the command:

```
awplus# show epsr counters
```

Related Commands
epsr mode master controlvlan primaryport
epsr mode transit controlvlan
show epsr

undebug epsr

This command applies the functionality of the no debug epsr command on page 57.3.

Part 7: Network Management

Chapter 58: NTP Introduction and Configuration

Introduction

This chapter describes the Network Time Protocol (NTP) service provided by the switch, and how to configure and monitor NTP on the switch.

NTP is a protocol for synchronizing the time clocks on a collection of network devices using a distributed client/server mechanism. NTP uses UDP (User Datagram Protocol) as the transport mechanism. NTP evolved from the Time Protocol (RFC 868) and the ICMP Timestamp message (RFC 792).

NTP provides protocol mechanisms to specify the precision and estimated error of the local clock and the characteristics of the reference clock to which it may be synchronized.

For detailed information about the commands used to configure NTP, see Chapter 59, NTP Commands.

Overview

NTP uses a subnetwork with primary reference clocks, gateways, secondary reference clocks, and local hosts. These are organized into a hierarchy with the more accurate clocks near the top and less accurate ones near the bottom.

A number of primary reference clocks, synchronized to national standards, are connected to widely accessible resources (such as backbone gateways or switches) operating as primary time servers. The primary time servers use NTP between them to crosscheck clocks, to mitigate errors due to equipment or propagation failures, and to distribute time information to local secondary time servers. The secondary time servers redistribute the time information to the remaining local hosts.

The hierarchical organization and distribution of time information reduces the protocol overhead, and allows selected hosts to be equipped with cheaper but less accurate clocks. NTP provides information which organizes this hierarchy on the basis of precision or estimated error.

- An NTP entity may be in one of the following operating modes; however, the switch's implementation of NTP supports two modes: client and server.
- An NTP entity operating in a client mode sends periodic messages to its peers, requesting synchronization by its peers.
- An NTP entity enters the server mode temporarily when it receives a client request message from one of its peers, and remains in server mode until the reply to the request has been transmitted.
- An NTP entity operating in symmetric active mode sends messages announcing its willingness to synchronize and be synchronized by its peers.
- An NTP entity enters symmetric passive mode in response to a message from a peer operating in Symmetric Active mode. An NTP entity operating in this mode announces its willingness to synchronize and be synchronized by its peers.
- An NTP entity operating in broadcast mode periodically sends messages announcing its willingness to synchronize all of its peers but not to be synchronized by any of them.

The same message format is used for both requests and replies. When a request is received, the server interchanges addresses and ports, fills in or overwrites certain fields in the message, recalculates the checksum, and returns it immediately. The information included in the NTP message allows each client/ server peer to determine the timekeeping characteristics of its peers, including the expected accuracies of their clocks. Each peer uses this information and selects the best time from possibly several other clocks, updates the local clock, and estimates its accuracy.

There is no provision in NTP for peer discovery, acquisition, or authentication. Data integrity is provided by the IP and UDP checksums. No reachability, circuit-management, duplicate-detection, or retransmission facilities are provided or necessary.

By its very nature clock synchronization requires long periods of time (hours or days) and multiple comparisons in order to maintain accurate timekeeping. The more comparisons performed, the greater the accuracy of the timekeeping.

NTP on the Switch

The implementation of NTP on the switch is based on the following RFCs:

- RFC 958, Network Time Protocol (NTP)
- RFC 1305, Network Time Protocol (Version 3) Specification, Implementation and Analysis
- RFC 1510, The Kerberos Network Authentication Service (V5)

Two modes of operation are supported: client and server. The switch is in client mode most of the time where it polls the configured peer at least once every preconfigured minimum time period.

The peer that the switch refers to must be a more accurate clock source than the switch itself or another switch directly connected to a more accurate clock source. The switch operates as a secondary time server. It cannot operate as a primary time server unless the primary clock source is operating in server mode. A primary clock source usually operates in broadcast mode, which is not supported by the switch's implementation of NTP. There is no support for clock selection or filtering. When the switch receives a valid reply from the peer, it synchronizes its own internal clock according to the information from the reply.

If the switch receives a synchronization request from an NTP client, it temporarily changes to server mode. It replies to the request with the current time from the switch's internal clock along with other information useful for synchronization. The switch's internal clock is accurate to 0.005 seconds.

Troubleshooting

Problem The switch is not assigning the time to devices on the LAN.

Solutions

- Check that the NTP peer's IP address is entered correctly.
- Check that the NTP peer can reach the switch, by pinging the switch from the NTP peer.

Problem The switch's clock does not synchronize with the NTP peer.

Solution

- The switch's clock can synchronize with the NTP peer only when its initial time is similar to the NTP peer's time (after setting the UTC offset). Manually set the switch's time so that it is approximately correct, and enable NTP again.
- Check that the UTC offset is correct.

Problem The switch's time is incorrect, even though it assigns the correct time to devices on the LAN.

Solution The UTC offset is probably incorrect, or needs to be adjusted for the beginning or end of summer time.

Configuration Example

NTP requires the IP module to be enabled and configured correctly.

The switch's implementation of NTP supports two modes: client and server mode. When a synchronization request is received from a client (e.g. a PC on a LAN), the switch enters server mode and responds with time information derived from the switch's own internal clock. Periodically the switch enters client mode, sending synchronization requests to a predefined peer to synchronize its own internal clock. The peer is assumed to be a primary clock source or another switch connected directly to a primary clock source.

This example illustrates how to configure two switches, one at a Head Office and one at a Regional Office, to provide a network time service. The Head Office switch is connected to a primary time server and provides the most accurate time information. The switch at the Regional Office uses the Head Office switch as its peer to avoid the cost of an additional WAN connection but provides slightly less accurate time information.

To configure NTP on the switch, the NTP module must be enabled and an NTP peer must be defined. NTP transfers time information in UTC format.

To set the switch to automatically change the time when summer time starts and ends, enable a summer time offset setting.

Example configuration parameters for a network time service:

Site	Regional Office	Head Office
Switch Name	RG1	HO1
IP Address of Switch	192.168.35.114	192.168.35.113
IP Address of Peer	192.168.35.113	192.168.13.3

Step 1: Enable NTP and define the NTP peer.

The NTP feature must be enabled on all switches that are to provide a network time service. Each switch must have a peer defined where the switch synchronizes its own internal clock. Enable NTP on the Head Office switch and specify a primary time server as the peer by using the commands:

```
        awplus# configure terminal
awplus(config)# ntp peer 192.168.13.3
```

Note that you can also specify an IPv6 address for an NTP peer:

```
        awplus# configure terminal
awplus(config)# ntp peer 2001:0db8:010d::1
```

Step 2: Configure the NTP parameters.

On each switch, the offset of local time from UTC time must be specified. In this example, both switches are in the same time zone, which is 12 hours ahead of UTC time. Use the following commands on both switches:

```
awplus(config)# clock timezone utc plus 12
```

Note that the range of offset is `<0-12>`.

Step 3: Check the NTP configuration.

Check the NTP configuration on each switch by using the command:

```
awplus# show ntp status
```

This command displays the following information on the Head Office switch.

```
Clock is synchronized, stratum 0, actual frequency is 0.0000
Hz, precision is 20 reference time is 00000000.00000000
(6:28:16.000 UTC Fri Feb  7 2036)clock offset is 0.000 msec,
root delay is 0.000 msec root dispersion is 0.000 msec,
```

Chapter 59: NTP Commands

Command List

This chapter provides an alphabetical reference for commands used to configure the Network Time Protocol (NTP). For more information, see Chapter 58, NTP Introduction and Configuration.

For information about modifying or redirecting the output from **show** commands to a file, see "Controlling "show" Command Output" on page 1.41.

ntp access-group

This command creates an NTP access group, and applies a basic IP access list to it. This allows you to control access to NTP services.

The **no** variant of this command removes the configured NTP access group.

Syntax

```
ntp access-group [peer|query-only|serve|serve-only]
    [<1-99>|<1300-1999>]

no ntp access-group [peer|query-only|serve|serve-only]
```

Parameter	Description
`peer`	Allows time requests and NTP control queries, and allows the system to synchronize itself to a system whose address passes the access list criteria.
`query-only`	Allows only NTP control queries from a system whose address passes the access list criteria.
`serve`	Allows time requests and NTP control queries, but does not allow the system to synchronize itself to a system whose address passes the access list criteria.
`serve-only`	Allows only time requests from a system whose address passes the access list criteria.
<1-99>	Standard IP access list.
<1300-1999>	Expanded IP access list.

Mode Global Configuration

Example To create an NTP peer access group for an extended IP access list, use the commands:

```
        awplus# configure terminal
awplus(config)# ntp access-group peer 1998
```

To disable the NTP peer access group created above, use the commands:

```
        awplus# configure terminal
awplus(config)# no ntp access-group peer
```

ntp authenticate

This command enables NTP authentication. This allows NTP to authenticate the associations with other systems for security purposes.

The **no** variant of this command disables NTP authentication.

Syntax

```
ntp authenticate
```

```
no ntp authenticate
```

Mode Global Configuration

Example To enable NTP authentication, use the commands:

```
        awplus#  configure terminal
awplus(config)#  ntp authenticate
```

To disable NTP authentication, use the commands:

```
        awplus#  configure terminal
awplus(config)#  no ntp authenticate
```

ntp authentication-key

This command defines each of the authentication keys. Each key has a key number, a type, and a value. Currently, the only key type supported is MD5.

The **no** variant of this disables the authentication key assigned previously using **ntp authentication-key**.

Syntax

```
ntp authentication-key <keynumber> md5 <key>

no ntp authentication-key <keynumber> md5 <key>
```

Parameter	Description
<keynumber>	*<1-4294967295>* The key number.
<key>	The authentication key.

Mode Global Configuration

Example To define an authentication key number `134343` and a key value `mystring`, use the commands:

```
         awplus# configure terminal
 awplus(config)# ntp authentication-key 134343 md5 mystring
```

To disable the authentication key number `134343` with the key value `mystring`, use the commands:

```
         awplus# configure terminal
 awplus(config)# no ntp authentication-key 134343 md5 mystring
```

ntp broadcastdelay

Use this command to set the estimated round-trip delay for broadcast packets.

Use the **no** variant of this command to reset the round-trip delay for broadcast packets to the default offset of 0 microseconds.

Syntax

```
ntp broadcastdelay <delay>

no ntp broadcastdelay
```

Parameter	Description
<delay>	*<1-999999>* The broadcast delay in microseconds.

Default 0 microsecond offset, which can only be applied with the **no** variant of this command.

Mode Global Configuration

Example To set the estimated round-trip delay to 23464 microseconds for broadcast packets, use these commands:

```
        awplus# configure terminal
awplus(config)# ntp broadcastdelay 23464
```

To reset the estimated round-trip delay for broadcast packets to the default setting (0 microseconds), use these commands:

```
        awplus# configure terminal
awplus(config)# no ntp broadcastdelay
```

ntp master

Use this command to make the device to be an authoritative NTP server, even if the system is not synchronized to an outside time source. Note that no stratum number is set by default.

Use the **no** variant of this command to stop the device being the designated NTP server.

Syntax

```
ntp master [<stratum>]

no ntp master
```

Parameter	Description
<stratum>	*<1-15>* The stratum number.

Mode Global Configuration

Usage The stratum number is null by default and must be set using this command. The stratum levels define the distance from the reference clock and exist to prevent cycles in the hierarchy. Stratum 1 is used to indicate time servers, which are more accurate than Stratum 2 servers.

Examples To stop the switch from being the designated NTP server use the commands:

```
        awplus# configure terminal
awplus(config)# no ntp master
```

To make the switch the designated NTP server with stratum number 2 use the commands:

```
        awplus# configure terminal
awplus(config)# ntp master 2
```

ntp peer

Use this command to configure an NTP peer association. An NTP association is a peer association if this system is willing to either synchronize to the other system, or allow the other system to synchronize to it.

Use the **no** variant of this command to remove the configured NTP peer association.

Syntax

```
ntp peer {<peeraddress>|<peername>}

ntp peer {<peeraddress>|<peername>}
   [prefer] [key <key>] [version <version>]

no ntp peer {<peeraddress>|<peername>}
```

Parameter	Description
<peeraddress>	Specify the IP address of the peer, entered in the form `A.B.C.D` for an IPv4 address, or in the form `X:X::X.X` for an IPv6 address.
<peername>	Specify the peer hostname. The peer hostname can resolve to an IPv4 and an IPv6 address.
`prefer`	Prefer this peer when possible.
`key` *<key>*	*<1-4294967295>* Configure the peer authentication key.
`version` *<version>*	*<1-4>* Configure for this NTP version.

Mode Global Configuration

Examples See the following commands for options to configure NTP peer association, key and NTP version for the peer with an IPv4 address of `192.0.2.23`:

```
        awplus# configure terminal
awplus(config)# ntp peer 192.0.2.23
awplus(config)# ntp peer 192.0.2.23 prefer
awplus(config)# ntp peer 192.0.2.23 prefer version 4
awplus(config)# ntp peer 192.0.2.23 prefer version 4 key 1234
awplus(config)# ntp peer 192.0.2.23 version 4 key 1234
awplus(config)# ntp peer 192.0.2.23 version 4
awplus(config)# ntp peer 192.0.2.23 key 1234
```

To remove an NTP peer association for this peer with an IPv4 address of `192.0.2.23`, use the following commands:

```
        awplus# configure terminal
awplus(config)# no ntp peer 192.0.2.23
```

See the following commands for options to configure NTP peer association, key and NTP version for the peer with an IPv6 address of `2001:0db8:010d::1`:

```
        awplus# configure terminal
awplus(config)# ntp peer 2001:0db8:010d::1
awplus(config)# ntp peer 2001:0db8:010d::1 prefer
awplus(config)# ntp peer 2001:0db8:010d::1 prefer version 4
awplus(config)# ntp peer 2001:0db8:010d::1 prefer version 4 key
                1234
awplus(config)# ntp peer 2001:0db8:010d::1 version 4 key 1234
awplus(config)# ntp peer 2001:0db8:010d::1 version 4
awplus(config)# ntp peer 2001:0db8:010d::1 key 1234
```

To remove an NTP peer association for this peer with an IPv6 address of `2001:0db8:010d::1`, use the following commands:

```
        awplus# configure terminal
awplus(config)# no ntp peer 2001:0db8:010d::1
```

ntp server

Use this command to configure an NTP server. This means that this system will synchronize to the other system, and not vice versa.

Use the **no** variant of this command to remove the configured NTP server.

Syntax

```
ntp server {<serveraddress>|<servername>}

ntp server {<serveraddress>|<servername>}
    [prefer] [key <key>] [version <version>]

no ntp server {<serveraddress>|<servername>}
```

Parameter	Description
`<serveraddress>`	Specify the IP address of the peer, entered in the form `A.B.C.D` for an IPv4 address, or in the form `X:X::X.X` for an IPv6 address.
`<servername>`	Specify the server hostname. The server hostname can resolve to an IPv4 and an IPv6 address.
`prefer`	Prefer this server when possible.
`key <key>`	*<1-4294967295>* Configure the server authentication key.
`version <version>`	*<1-4>* Configure for this NTP version.

Mode Global Configuration

Examples See the following commands for options to configure an NTP server association, key and NTP version for the server with an IPv4 address of `192.0.1.23`:

```
        awplus# configure terminal
awplus(config)# ntp server 192.0.1.23
awplus(config)# ntp server 192.0.1.23 prefer
awplus(config)# ntp server 192.0.1.23 prefer version 4
awplus(config)# ntp server 192.0.1.23 prefer version 4 key 1234
awplus(config)# ntp server 192.0.1.23 version 4 key 1234
awplus(config)# ntp server 192.0.1.23 version 4
awplus(config)# ntp server 192.0.1.23 key 1234
```

To remove an NTP peer association for this peer with an IPv4 address of `192.0.1.23`, use the following commands:

```
        awplus# configure terminal
awplus(config)# no ntp server 192.0.1.23
```

See the following commands for options to configure an NTP server association, key and NTP version for the server with an IPv6 address of `2001:0db8:010e::2`:

```
        awplus# configure terminal

awplus(config)# ntp server 2001:0db8:010e::2

awplus(config)# ntp server 2001:0db8:010e::2 prefer

awplus(config)# ntp server 2001:0db8:010e::2 prefer version 4

awplus(config)# ntp server 2001:0db8:010e::2 prefer version 4
                key 1234

awplus(config)# ntp server 2001:0db8:010e::2 version 4 key 1234

awplus(config)# ntp server 22001:0db8:010e::2 version 4

awplus(config)# ntp server 2001:0db8:010e::2 key 1234
```

To remove an NTP peer association for this peer with an IPv6 address of `2001:0db8:010e::2`, use the following commands:

```
        awplus# configure terminal

awplus(config)# no ntp server 2001:0db8:010e::2
```

ntp trusted-key

This command defines a list of trusted authentication keys. If a key is trusted, this system will be ready to synchronize to a system that uses this key in its NTP packets.

Use the **no** variant of this command to remove a configured trusted authentication key.

Syntax

```
ntp trusted-key <1-4294967295>

no ntp trusted-key <1-4294967295>
```

Parameter	Description
<1-4294967295>	The specific key number.

Mode Global Configuration

Example To define a trusted authentication key numbered 234675, use the following commands:

```
        awplus# configure terminal
awplus(config)# ntp trusted-key 234676
```

To remove the trusted authentication key numbered 234675, use the following commands:

```
        awplus# configure terminal
awplus(config)# no ntp trusted-key 234676
```

show counter ntp

This command displays packet counters for NTP.

Syntax `show counter ntp`

Mode User Exec and Privileged Exec

Output Figure 59-1: Example output from the **show counter ntp** command

```
NTP counters
Pkts Sent              ......... 0
Pkts Received          ......... 70958
Pkts Processed         ......... 0
Pkts current version   ......... 0
Pkts old version       ......... 0
Pkts unknown version   ......... 0
Pkts access denied     ......... 70958
Pkts bad length        ......... 0
Pkts bad auth          ......... 0
Pkts rate exceed       ......... 0
```

Table 59-1: Parameters in the output from the **show counter ntp** command

Parameter	Description
`Pkts Sent`	Total number of NTP client and server packets sent by your device.
`Pkts Received`	Total number of NTP client and server packets received by your device.
`Pkts Processed`	The number of packets processed by NTP. NTP processes a packet once it has determined that the packet is valid by checking factors such as the packet's authentication, format, access rights and version.
`Pkts current version`	The number of version 4 NTP packets received.
`Pkts old version`	The number of NTP packets received that are from an older version, down to version 1, of NTP. NTP is compatible with these versions and processes these packets.
`Pkts unknown version`	The number of NTP packets received that are an earlier version than version 1, or a higher version than version 4. NTP cannot process these packets.
`Pkts access denied`	The number of NTP packets received that do not match any access list statements in the NTP access-groups. NTP drops these packets.
`Pkts bad length`	The number of NTP packets received that do not conform to the standard packet length. NTP drops these packets.
`Pkts bad auth`	The number of NTP packets received that failed authentication. NTP drops these packets. Packets can only fail authentication if NTP authentication is enabled with the ntp authenticate command.
`Pkts rate exceed`	The number of packets dropped because the packet rate exceeded its limits.

Example To display counters for NTP, use the command:

```
awplus# show counter ntp
```

show ntp associations

Use this command to display the status of NTP associations. Use the detail option for displaying detailed information about the associations.

Syntax `show ntp associations [detail]`

Mode User Exec and Privileged Exec

Example See the sample output of the **show ntp associations** and **show ntp associations detail** commands displaying the status of NTP associations

Figure 59-2: Example output from the **show ntp associations** command

```
awplus#show ntp associations
  address         ref clock       st  when  poll reach   delay  offset    disp
 ~192.0.2.23      INIT            16     -   512   000     0.0     0.0     0.0
 * master (synced), # master (unsynced), + selected, - candidate, ~ configured
awplus#
```

Figure 59-3: Example output from the **show ntp associations detail** command

```
awplus#show ntp associations detail
192.0.2.23 configured, sane, valid, leap_sub, stratum 16
ref ID INIT, time 00000000.00000000 (06:28:16.000 UTC Thu Feb  7 2036)
our mode client, peer mode unspec, our poll intvl 512, peer poll intvl 1024
root delay 0.00 msec, root disp 0.00, reach 000,
delay 0.00 msec, offset 0.0000 msec, dispersion 0.00
precision 2**-19,
org time 00000000.00000000 (06:28:16.000 UTC Thu Feb  7 2036)
rcv time 00000000.00000000 (06:28:16.000 UTC Thu Feb  7 2036)
xmt time cf11f2a4.cedde5e4 (00:39:00.808 UTC Tue Feb  2 2010)
filtdelay =  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00
filtoffset =  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00
filterror =  16000.00  16000.00  16000.00  16000.00  16000.00  16000.00  16000.0
0  16000.00
```

Table 59-2: Parameters in the output from the **show ntp associations** command

Parameter	Description
`address`	Peer IP address
`ref clock`	IP address for reference clock
`st`	Stratum. The number of hops between the server and the accurate time source.
`poll`	Time between NTP requests from the device to the server.
`reach`	Shows whether or not the NTP server responded to the last request.
`delay`	Round trip delay between the device and the server.
`offset`	Difference between the device clock and the server clock.
`disp`	Lowest measure of error associated with peer offset based on delay.

show ntp status

Use this command to display the status of the Network Time Protocol (NTP).

Syntax `show ntp status`

Mode User Exec and Privileged Exec

Example See the sample output of the **show ntp status** command displaying information about the Network Time Protocol.

Figure 59-4: Example output from the **show ntp status** command

```
awplus#sh ntp status
Clock is synchronized, stratum 3, reference is 127.127.1.0
actual frequency is 0.0000 Hz, precision is 2**-19
reference time is cf11f3f2.c7c081a1 (00:44:34.780 UTC Tue Feb  2 
2010)
clock offset is 0.000 msec, root delay is 0.000 msec
root dispersion is 7947729.000 msec,
awplus#
```

Chapter 60: Dynamic Host Configuration Protocol (DHCP) Introduction

Introduction

This chapter describes the Dynamic Host Configuration Protocol (DHCP) support provided by your device. This includes how to configure your device to:

- act as a DHCP and BOOTP server
- act as a DHCP relay agent
- use the DHCP client to obtain IP addresses for its own interfaces

Note that you can configure your device to operate as both a DHCP relay agent and a DHCP/ BOOTP server.

BOOTP

Bootstrap Protocol (BOOTP) is a UDP-based protocol that enables a booting host to dynamically configure itself without external interventions. A BOOTP server responds to requests from BOOTP clients for configuration information, such as the IP address the client should use. BOOTP is defined in RFC 951, Bootstrap Protocol (BOOTP).

RFC 1542, Clarifications and Extensions for the Bootstrap Protocol, defines extensions to the BOOTP protocol, including the behavior of a DHCP relay agent.

DHCP

DHCP is widely used to dynamically assign host IP addresses from a centralized server that reduces the overhead of administrating IP addresses. DHCP helps conserve the limited IP address space because IP addresses no longer need to be permanently assigned to hosts. DHCP centrally manages IP address assignment for a large number of subscribers.

DHCP is based on BOOTP, and is defined in RFC 2131. It extends the BOOTP mechanism by providing:

- a method for passing configuration information to hosts on a TCP/IP network
- automatic allocation of reusable network addresses
- other additional configuration options

When your device is configured as a DHCP server, it allocates IP addresses and other IP configuration parameters to clients (hosts), when the client requests them. This lets you configure your IP network without manually configuring every client. Note that each client must also be configured to receive its IP address automatically.

As well as addresses, a DHCP server assigns a wide range of parameters to clients, including subnet information and mask, domain and hostname, server addresses, keepalive times, MTUs, boot settings, encapsulation settings, time settings, and TCP settings.

DHCP is designed to interoperate with BOOTP clients and DHCP clients, without the BOOTP clients needing any change to their initialization software.

DHCP Relay Agents

DHCP relay agents pass BOOTP and DHCP messages between servers and clients. Networks where the DHCP or BOOTP server does not reside on the same IP subnet as its clients need the intermediate routers to act as relay agents. A maximum number of 400 DHCP relay agents (one per interface) can be configured on the device. Once this limit has been reached, any further attempts to configure DHCP relay agents will not be successful.

Configuring the DHCP Server

The DHCP server uses **address pools** when responding to DHCP client requests. Address pools contains specific IP configuration details that the DHCP server can allocate to a client. You can configure multiple address pools on the device for different networks.

To configure a pool, you must:

- Create the Pool and enter its configuration mode.
- Define the Network the pool applies to.
- Define the Range of IP addresses that the server can allocate to clients. You can specify multiple address ranges for each pool.
- Set the Lease for the clients. This defines whether the clients receive a dynamic, permanent, or static IP address.
- Set the Options (standard and user-defined) that the clients of a pool require when configuring their IP details.

After configuring the address pools, you can then enable the DHCP server by using the command:

```
awplus(config)# service dhcp-server
```

For networks where you do not want the server to respond to BOOTP requests, you can configure the DHCP server so that it ignores them, by using the command:

```
awplus(config)# ip dhcp bootp ignore
```

Create the Pool

A DHCP pool is identified by a name. To create a DHCP pool and enter the configuration mode for the pool, use the command:

```
awplus(config)# ip dhcp pool <pool-name>
```

Define the Network

Define the network that the DHCP clients are in. You can define one network per address pool. Use the following command to define the network after defining the DHCP pool first:

```
awplus(dhcp-config)# network
```

- For remote clients, set the network address to the network of the remote clients. The **network** command does not need to match a specific interface's network, because the DHCP server listens on all IP interfaces for DHCP requests.
- For locally connected clients, ensure that the desired interface has an IP address and subnet mask defined; use the **ip address IPADDR** command to set a static address. Enter the configuration mode for the pool, and set the DHCP address pool's network to match the interface's network. Pools that span multiple interfaces are possible only if the interface networks are contiguous.

Define the Range

Configure an IP address range for the pool. This range must be in the same subnet as the pool's network setting. Use the command:

```
awplus(dhcp-config)# range <ip-address> [<ip-address>]
```

The first IPv4 address specifies the **low end of the range, while the second IP address is the high end.** You can set the range to a single IP address by specifying only one IP address.

Set the Lease

The DHCP server assigns IP settings to hosts for specific times (the lease time). Each DCHP pool has one lease time setting. You can use DHCP to allocate the following types of addresses:

- A **dynamic** IP addresses
  These are available to a host for a limited amount of time. When the lease expires, the server can reallocate the IP address to another device. To set the lease time for the DHCP pool so that it assigns dynamic IP addresses, use the command:

  ```
  awplus(dhcp-config)# lease <days> <hours> <minutes>
                       [<seconds>]
  ```

- A **permanent** IP addresses
  These are available to a host for an unlimited amount of time. To set the lease time to assign permanent IP addresses, use the command:

  ```
  awplus(dhcp-config)# lease infinite
  ```

- A **static** IP addresses
  These are allocated to a particular client. The DHCP server recognizes the client by its MAC address. This lets you use DHCP to manage most of your network automatically, while having unchanging IP addresses on key devices such as servers. To assign a static IP address to a device, use the command:

  ```
  awplus(dhcp-config)# host <ip-address> <mac-address>
  ```

BOOTP requests can be satisfied by pools with leases set to infinity.

Enable DHCP Leasequery

The DHCP Leasequery protocol (RFC 4388) allows a device or process, for example a DHCP relay agent, to obtain IP address information directly from the DHCP server using DHCPLEASEQUERY messages.

DHCPLEASEQUERY messages support three query regimes:

- IP address

  Only an IP address is supplied in the DHCPLEASEQUERY message. The DHCP server will return any information that it has on the most recent client to have been assigned that IP address.

- MAC address

  Only a MAC address is supplied in the DHCPLEASEQUERY message. The DHCP server will return any information that it has on the IP address most recently accessed by a client with that MAC address. Also, the DHCP server may supply additional IP addresses that have been associated with that MAC address in different subnets.

- Client identifier option

  Only a Client identifier option is supplied in the DHCPLEASEQUERY message. The DHCP server will return any information that it has on the IP address most recently accessed by a client with that Client identifier. Also, the DHCP server may supply additional IP addresses that have been associated with Client identifier in different subnets.

An AlliedWare Plus DHCP server implementing DHCP Leasequery supports all three query regimes.

If the DHCP Leasequery feature is enabled, when a DHCP relay agent needs to know the location of an IP endpoint and sends a DHCPLEASEQUERY message, the DHCP server will reply with either a DHCPLEASEACTIVE, DHCPLEASEUNASSIGNED, or DHCPLEASEUNKNOWN message.

When the DHCP server replies to a DHCPLEASEQUERY message:

- a DHCPLEASEACTIVE message allows the DHCP relay agent to determine the IP endpoint location and the remaining duration of the IP address lease
- a DHCPLEASEUNASSIGNED message indicates that there is no current active lease for the IP address, but the DHCP server does manage that IP address
- a DHCPLEASEUNKNOWN message indicates that the DHCP server supports DHCP Leasequery but has no knowledge of the query information specified in the DHCPLEASEQUERY message (e.g., IP address, MAC address, or Client identifier option)

To enable the DHCP Leasequery feature, use the commands:

```
        awplus# configure terminal
awplus(config)# ip dhcp leasequery enable
```

To disable the DHCP Leasequery feature, use the commands:

```
        awplus# configure terminal
awplus(config)# no ip dhcp leasequery enable
```

To display information about DHCP Leasequery messages, use either of the commands:

```
awplus# show counter dhcp-server
awplus# show ip dhcp server statistics
```

To display information about the current configuration of the DHCP server, including whether the DHCP server is configured to support DHCP Leasequery, use the command:

```
awplus# show ip dhcp server summary
```

Set the Options

DHCP allows clients to receive options from the DHCP server. Options describe the network configuration, and various services that are available on the network. Options are configured separately on each DHCP pool. You can configure both standard predefined options and user-defined options for a DHCP pool.

To create a user-defined option, use the commands:

```
        awplus# configure terminal
awplus(config)# ip dhcp option <1-254> [name <option-
                name>] [<option-type>]
```

To add a user-defined option to a DHCP address pool, use the command sequence:

```
     awplus(config)# ip dhcp pool <pool-name>
awplus(dhcp-config)# option [<1-254>|<option-name>]
                     <option-value>
```

It is possible to add a user-defined option with the same number as an existing pre-defined option. If this situation occurs, the user-defined option takes precedence—that is, it overrides but does not eliminate the standard option.

You can set some pre-defined options using the following commands:

To set a subnet mask (option 1) for the address pool, use the command:

```
awplus(dhcp-config)# subnet-mask <mask>
```

To add a domain name (option 15) for the address pool, use the command:

```
awplus(dhcp-config)# domain-name <domain-name>
```

To add a default router (option 3) for the address pool, use the command:

```
awplus(dhcp-config)# default-router <ip-address>
```

To add a DNS server (option 6) for the address pool, use the command:

```
awplus(dhcp-config)# dns-server <ip-address>
```

DHCP Lease Probing

Probing is used by the DHCP server to check whether an IP address it wants to lease to a client is already being used by another host. Probing is configured on a per-DHCP pool basis. You can specify probing either by ICMP Echo Request (ping) or by ARPing. ARP probing is useful in networks where ICMP may be blocked on some devices, whereas ARP is always supported. ARP and ping probing are mutually exclusive and cannot operate concurrently within a DHCP pool.

Probing is enabled by default when a DHCP pool is created.

To enable probing if probing has previously been disabled for a DHCP pool, enter the configuration mode for the pool with the ip dhcp pool command and then use the command:

```
awplus(dhcp-config)# probe enable
```

The default probe type is ping. To specify the probe type as ARP, enter the configuration mode for the pool and then use the command:

```
awplus(dhcp-config)# probe type arp
```

To set the timeout value in milliseconds to wait for a response after each probe packet is sent, use the command:

```
awplus(dhcp-config)# probe timeout <50-5000>
```

To specify the number of packets sent for each lease probe, use the command:

```
awplus(dhcp-config)# probe packets <0-10>
```

To disable probing for a DHCP pool, enter the configuration mode for the pool and then use the command:

```
awplus(dhcp-config)# no probe enable
```

To display the lease probe configuration settings for a specific DHCP pool or for all DHCP pools configured on the device, use the command:

```
awplus# show ip dhcp pool [<address-pool>]
```

DHCP Relay Agent Introduction

DHCP relay agents pass BOOTP messages between servers and clients. Networks where the DHCP or BOOTP server does not reside on the same IP subnet as its clients need the routers attached to the subnet to act as DHCP relay agents.

Note that both BOOTP and DHCP use BOOTP messages, allowing DHCP relay agents to relay all their packets.

Your device's DHCP Relay Agent relays these message types:

- BOOTREQUEST messages originating from any of the device's interfaces to a user-defined destination
- BOOTREPLY messages addressed to BOOTP clients on networks directly connected to the device

The relay agent ignores BOOTREPLY messages addressed to clients on networks not directly connected to the device. The device treats these as ordinary IP packets for forwarding.

A BOOTREQUEST message may be relayed via unicast, multicast or broadcast methods. In the last case, the message does not re-broadcast to the interface from which it was received. The relay destinations are configured independently of other broadcast forwarders' destinations (e.g. TFTP).

The hops field in a BOOTP message records the number of hops (routers) the message has been through. If the value of the hops field exceeds a predefined threshold, the relay agent discards the message.

Configuring the DHCP Relay Agent

To enable the DHCP relay agent on your device, use the commands:

```
        awplus# configure terminal
awplus(config)# service dhcp-relay
```

You must define a relay destination on one of the device's interfaces before the relay agent can relay packets. This is the path to the DHCP server. To define a relay destination, use the commands:

```
   awplus(config)# interface <interface-name>
awplus(config-if)# ip dhcp-relay server-address <ip-address>
```

You can define more than one relay destination on your device. The following table describes how the relay agent forwards the packets.

If an interface has...	Then the relay agent relays BOOTP packets it receives on that interface to...
one relay destination defined	the relay destination.
multiple relay destinations defined	each defined relay destination.

To delete a relay destination, use the command:

```
awplus(config-if)# no ip dhcp-relay server-address <ip-address>
```

See the ip dhcp-relay server-address command on page 61.22 and the service dhcp-relay command on page 61.35 for detailed command description and command examples. DHCP servers with IPv4 and IPv6 addresses can now be configured with ip dhcp-relay server-address.

When the 'hops' field in a BOOTP message exceeds a predefined threshold the BOOTP message is discarded. The default of the threshold is 10. To set the threshold, use the command:

```
awplus(config-if)# ip dhcp-relay maxhops <1-255>
```

To display the current configuration of the DHCP relay agent, use the command:

```
awplus# show ip dhcp-relay [interface <interface-name>]
```

DHCP Relay Agent Option 82

Enabling the DCHP Option 82 feature on the switch allows the switch to insert extra information into the DHCP packets that it is relaying. This information enables more accurate identification of a subscriber, as it states which switch port on which relay switch the subscriber is connected to. The information is stored in a specific optional field in the DHCP packet, namely, the agent-information field, which has option ID 82.

The DHCP relay agent inserts the Option 82 information into the DHCP packets that it is relaying to a DHCP server. DHCP servers that are configured to recognize Option 82 may use the information to implement IP addresses, or other parameter assignment policies, based on the network location of the client device. Alternatively, the server can simply log this information to create a detailed audit trail of the locations of the clients to which given addresses were allocated at given times.

If the DHCP Relay Agent Option 82 feature is enabled, the DHCP packet flow is as follows:

- The DHCP client generates a DHCP request and broadcasts it on the network.
- The DHCP relay agent intercepts the broadcast DHCP request packet and inserts the relay agent information option (Option 82) in the packet.
- The DHCP relay agent forwards the DHCP request that includes the Option 82 field to the DHCP server.
- The DHCP server receives the packet.
- If the DHCP server supports Option 82, then it echoes the Option 82 field in the DHCP reply. If the server does not support Option 82, it ignores the option and does not echo it in the reply.
- The DHCP server unicasts the reply to the relay agent.
- The relay agent removes the Option 82 field and forwards the packet to the switch port connected to the DHCP client that sent the DHCP request.

For more information about DHCP Relay Agent Option 82, see RFC 3046. Option 82 can be:

- added to packets relayed from the DHCP client to DHCP server
- removed from packets relayed from DHCP server to DHCP client
- checked from sources closer to the client

To enable the relay agent to insert its details into the Option 82 field in requests received from clients attached to a particular interface, use the command:

```
   awplus(config)# interface <interface-name>
awplus(config-if)# ip dhcp-relay agent-option
```

This applies to requests received with no other agent relay information in the Option 82 field.

The Option 82 field contains sub-options. You can specify a value for the Remote ID sub-option, which contains information that identifies the host. To specify a value for the Remote ID, use the command:

```
   awplus(config)# interface <interface-name>
awplus(config-if)# ip dhcp-relay agent-option remote-id
                   <remote-id>
```

If a Remote ID value is not specified, the Remote ID sub-option is set to the switch's MAC address.

Note that the Option 82 agent information added by DHCP Relay differs from the information inserted by the DHCP snooping (see "DHCP Option 82" on page 52.4).

Dealing with client-originated packets that already contain Option 82 information

The discussion above deals with the case where the DHCP requests arriving from the clients do not already contain Option 82 information. However, it is possible that the requests arriving from the clients to the relay agent could already contain Option 82 information. There are two main circumstances in which this can occur:

1. A client is maliciously inserting bogus information into the packet in an attempt to subvert the process of identifying the client's location

2. A Layer 2 DHCP snooping switch, that sits between the clients and the DCHP relay, is validly inserting the Option 82 information into the packets. The DHCP snooping switch is not acting as a relay agent, so it is not filling in the **giaddr** field (the relay IP address field) in the packet; it is only inserting the Option 82 information.

In case 1, you would want to drop the packets that contain the bogus information (or, at least remove the bogus information). In case 2, you would want to forward the valid information to the DHCP server.

To configure the switch to check for the presence of Option 82 information in incoming DHCP requests, configure DHCP-relay agent-option checking, with the command (in Interface Configuration mode):

```
   awplus(config)# interface <interface-name>
awplus(config-if)# ip dhcp-relay agent-option checking
```

By default, this will cause the switch to act as follows:

- If the incoming DHCP request has a null IP address (0.0.0.0) in the **giaddr** field, and contains Option 82 information, drop the packet. This assumes that such a packet has been maliciously created by a client.

- If an incoming DHCP request has a non-null in the **giaddr** field, and contains Option 82 information, then replace the Option 82 field with the current switch's own information. This assumes that a non-null giaddr field indicates that the packet has already passed through a valid DHCP relay device, and so the presence of the Option 82 information is not an indication of malicious intent.

The action taken on packets that have a null giaddr field and an Option 82 field present cannot be altered once the agent-option check has been enabled. But the action taken on packets with a non-null giaddr field and an Option 82 field is configurable. The command to configure this action is shown below:

```
   awplus(config)# interface <interface-name>
awplus(config-if)# ip dhcp-relay information policy
```

This command takes parameters that can configure the switch to:

- Leave the existing Option 82 field untouched
- Append its own Option 82 field after the existing field
- Drop the packet
- Replace the existing Option 82 information with its own (the default).

DHCP Relay Agent Option 82 maximum message length

Where a DHCP relay (that has Option 82 insertion enabled) receives a request packet from a DHCP client, it will append the Option 82 component data, and forward the packet to the DHCP server. The DHCP client will sometimes issue packets containing pad option fields that can be overwritten with Option 82 data. Where there are insufficient pad option fields to contain all the Option 82 data, the DHCP relay will increase the packet size to accommodate the Option 82 data. If the new (increased) packet size exceeds that defined by the **maximum-message-length** parameter, of the ip dhcp-relay max-message-length command then the DHCP relay will drop the packet.

```
   awplus(config)# interface <interface-name>
awplus(config-if)# ip dhcp-relay max-message-length 1200
```

Configuring the DHCP Client

You can configure an interface on your device with a static IP address, or with a dynamic IP address assigned using your device's DHCP client. When you use the DHCP client, it obtains the IP address for the interface, and other IP configuration parameters, from a DHCP server. To configure an interface and gain its IP configuration using the DHCP client, use the command:

```
   awplus(config)#  interface <ifname>
awplus(config-if)#  ip address dhcp [client-id <interface>]
                    [hostname <hostname>]
```

The DHCP client supports the following IP configuration options:

- Option 1—the subnet mask for your device.
- Option 3—a list of default routers.
- Option 6—a list of DNS servers. This list appends the DNS servers set on your device with the ip name-server command.
- Option 15—a domain name used to resolve host names. This option replaces the domain name set with the ip domain-name command. Your device ignores this domain name if it has a domain list set using the ip domain-list command.
- Option 51—lease expiration time.

If an IP interface is configured to get its IP address and subnet mask from DHCP, the interface does not take part in IP routing until the IP address and subnet mask have been set by DHCP.

For information on configuring a static IP address on an interface, see the ip address command on page 25.14.

Clearing Dynamically Allocated Lease Bindings

A lease binding is the mapping of an IP address to a physical address. To clear dynamically allocated lease bindings, use the command:

```
awplus#  clear ip dhcp binding {ip <ip-address>|
         mac <mac-address>|all|pool <pool-name>|
         range <low-ip-address> <high-ip-address>}
```

You have the option to clear either a specific lease binding, specified by IP or MAC address, or to clear several lease bindings at once. The options for clearing multiple lease bindings are:

- **all**, to clear all DHCP bindings
- **pool**, to clear a specific DHCP server address pool
- **range**, to clear a range of DHCP clients

Chapter 61: Dynamic Host Configuration Protocol (DHCP) Commands

Command List

This chapter provides an alphabetical reference for commands used to configure DHCP. For more information, see Chapter 60, Dynamic Host Configuration Protocol (DHCP) Introduction.

For information about modifying or redirecting the output from **show** commands to a file, see "Controlling "show" Command Output" on page 1.41.

bootfile

This command sets the boot filename for a DHCP server pool. This is the name of the boot file that the client should use in its bootstrap process. It may need to include a path.

The **no** variant of this command removes the boot filename from a DHCP server pool.

Syntax

```
bootfile <filename>

no bootfile
```

Parameter	Description
`<filename>`	The boot file name.

Mode DHCP Configuration

Example To configure the boot filename for a pool `P2`, use the command:

```
           awplus# configure terminal
   awplus(config)# ip dhcp pool P2
awplus(dhcp-config)# bootfile boot/main_boot.bt
```

clear ip dhcp binding

This command clears either a specific lease binding or the lease bindings specified by the command. The command will only take effect on dynamically allocated bindings, not statically configured bindings.

Syntax

```
clear ip dhcp binding {ip <ip-address>|mac <mac-address>|all|pool
    <pool-name>|range <low-ip-address> <high-ip-address>}
```

Parameter	Description
`ip <ip-address>`	IPv4 address of the DHCP client, in dotted decimal notation in the format A.B.C.D.
`mac <mac-address>`	MAC address of the DHCP client, in hexadecimal notation in the format HHHH.HHHH.HHHH.
`all`	All DHCP bindings.
`pool <pool-name>`	Description used to identify DHCP server address pool. Valid characters are any printable character. If the name contains spaces then you must enclose these in "quotation marks".
`range <low-ip-address> <high-ip-address>`	IPv4 address range for DHCP clients, in dotted decimal notation. The first IP address is the low end of the range, the second IP address is the high end of the range.

Mode Privileged Exec

Usage A specific binding may be deleted by **ip** address or **mac** address, or several bindings may be deleted at once using **all**, **pool** or **range**.

Note that if you specify to clear the **ip** or **mac** address of what is actually a static DHCP binding, an error message is displayed. If **all**, **pool** or **range** are specified and one or more static DHCP bindings exist within those addresses, any dynamic entries within those addresses are cleared but any static entries are not cleared.

Examples To clear the specific IP address binding `192.168.1.1`, use the command:

```
awplus# clear ip dhcp binding ip 192.168.1.1
```

To clear all dynamic DHCP entries, use the command:

```
awplus# clear ip dhcp binding all
```

Related Commands show ip dhcp binding

default-router

This command adds a default router to the DHCP address pool you are configuring. You can use this command multiple times to create a list of default routers on the client's subnet. This sets the router details using the pre-defined option 3. Note that if you add a user-defined option 3 using the **option** command, then you will override any settings created with this command.

The **no** variant of this command removes either the specified default router, or all default routers from the DHCP pool.

Syntax

```
default-router <ip-address>

no default-router [<ip-address>]
```

Parameter	Description
<ip-address>	IPv4 address of the default router, in dotted decimal notation.

Mode DHCP Configuration

Examples To add a router with an IP address 192.168.1.2 to the DHCP pool named P2, use the following commands:

```
          awplus# configure terminal
  awplus(config)# ip dhcp pool P2
awplus(dhcp-config)# default-router 192.168.1.2
```

To remove a router with an IP address 192.168.1.2 to the DHCP pool named P2, use the following commands:

```
          awplus# configure terminal
  awplus(config)# ip dhcp pool P2
awplus(dhcp-config)# no default-router 192.168.1.2
```

To remove all routers from the DHCP pool named P2, use the following commands:

```
          awplus# configure terminal
  awplus(config)# ip dhcp pool P2
awplus(dhcp-config)# no default-router
```

dns-server

This command adds a Domain Name System (DNS) server to the DHCP address pool you are configuring. You can use this command multiple times to create a list of DNS name servers available to the client. This sets the DNS server details using the pre-defined option 6. Note that if you add a user-defined option 6 using the option command on page 61.28, then you will override any settings created with this command.

The **no** variant of this command removes either the specified DNS server, or all DNS servers from the DHCP pool.

Syntax

```
dns-server <ip-address>

no dns-server [<ip-address>]
```

Parameter	Description
<ip-address>	IPv4 address of the DNS server, in dotted decimal notation.

Mode DHCP Configuration

Examples To add the DNS server with the assigned IP address 192.168.1.1 to the DHCP pool named P1, use the following commands:

```
            awplus# configure terminal
    awplus(config)# ip dhcp pool P2
awplus(dhcp-config)# dns-server 192.168.1.1
```

To remove the DNS server with the assigned IP address 192.168.1.1 from the DHCP pool named P1, use the following commands:

```
            awplus# configure terminal
    awplus(config)# ip dhcp pool P2
awplus(dhcp-config)# no dns-server 192.168.1.1
```

To remove all DNS servers from the DHCP pool named P1, use the following commands:

```
            awplus# configure terminal
    awplus(config)# ip dhcp pool P2
awplus(dhcp-config)# no dns-server
```

Related Commands default-router
option
service dhcp-server
show ip dhcp pool
subnet-mask

domain-name

This command adds a domain name to the DHCP address pool you are configuring. Use this command to specify the domain name that a client should use when resolving host names using the Domain Name System. This sets the domain name details using the pre-defined option 15. Note that if you add a user-defined option 15 using the option command on page 61.28, then you will override any settings created with this command.

The **no** variant of this command removes the domain name from the address pool.

Syntax

```
domain-name <domain-name>
```

```
no domain-name
```

Parameter	Description
<domain-name>	The domain name you wish to assign the DHCP pool. Valid characters are any printable character. If the name contains spaces then you must enclose it in "quotation marks".

Mode DHCP Configuration

Examples To add the domain name `Nerv_Office` to DHCP pool `P2`, use the commands:

```
             awplus# configure terminal
     awplus(config)# ip dhcp pool P2
awplus(dhcp-config)# domain-name Nerv_Office
```

To remove the domain name `Nerv_Office` from DHCP pool `P2`, use the commands:

```
             awplus# configure terminal
     awplus(config)# ip dhcp pool P2
awplus(dhcp-config)# no domain-name Nerv_Office
```

Related Commands

default-router
dns-server
option
service dhcp-server
show ip dhcp pool
subnet-mask

host

This command adds a static host address to the DHCP address pool you are configuring. The client with the matching MAC address is permanently assigned this IP address. No other clients can request it.

The **no** variant of this command removes the specified host address from the DHCP pool. Use the **no host all** command to remove all static host addresses from the DHCP pool.

Syntax

```
host <ip-address> <mac-address>

no host <ip-address>

no host all
```

Parameter	Description
<ip-address>	IPv4 address of the DHCP client, in dotted decimal notation in the format A.B.C.D
<mac-address>	MAC address of the DHCP client, in hexadecimal notation in the format HHHH.HHHH.HHHH

Mode DHCP Configuration

Usage Note that a network/mask must be configured using a **network** command before issuing a **host** command. Also note that a host address must match a network to add a static host address.

Examples To add the host at 192.168.1.5 with the MAC address 000a.451d.6e34 to DHCP pool 1, use the commands:

```
             awplus# configure terminal
     awplus(config)# ip dhcp pool 1
awplus(dhcp-config)# network 192.168.1.0/24
awplus(dhcp-config)# host 192.168.1.5 000a.451d.6e34
```

To remove the host at 192.168.1.5 with the MAC address 000a.451d.6e34 from DHCP pool 1, use the commands:

```
             awplus# configure terminal
     awplus(config)# ip dhcp pool 1
awplus(dhcp-config)# no host 192.168.1.5 000a.451d.6e34
```

Related Commands

lease
range
show ip dhcp pool

ip address dhcp

This command activates the DHCP client on the interface you are configuring. This allows the interface to use the DHCP client to obtain its IP configuration details from a DHCP server on its connected network.

The **client-id** and **hostname** parameters are identifiers that you may want to set in order to interoperate with your existing DHCP infrastructure. If neither option is needed, then the DHCP server uses the MAC address field of the request to identify the host.

The DHCP client supports the following IP configuration options:

- Option 1 - the subnet mask for your device.
- Option 3 - a list of default routers.
- Option 6 - a list of DNS servers. This list appends the DNS servers set on your device with the ip name-server command.
- Option 15 - a domain name used to resolve host names. This option replaces the domain name set with the ip domain-name command. Your device ignores this domain name if it has a domain list set using the ip domain-list command.
- Option 51 - lease expiration time.

The **no** variant of this command stops the interface from obtaining IP configuration details from a DHCP server.

Syntax

```
ip address dhcp [client-id <interface>] [hostname <hostname>]

no ip address dhcp
```

Parameter	Description
<interface>	The name of the interface you are activating the DHCP client on. If you specify this, then the MAC address associated with the specified interface is sent to the DHCP server in the optional identifier field. Default: no default
<hostname>	The hostname for the DHCP client on this interface. Typically this name is provided by the ISP. Default: no default

Mode Interface Configuration for a VLAN interface.

Examples To set the interface `vlan10` to use DHCP to obtain an IP address, use the command:

```
           awplus# configure terminal
   awplus(config)# interface vlan10
awplus(config-if)# ip address dhcp
```

To stop the interface `vlan10` from using DHCP to obtain its IP address, use the command:

```
           awplus# configure terminal
   awplus(config)# interface vlan10
awplus(config-if)# no ip address dhcp
```

Related Commands	ip address
Validation Commands	show running-config show running-config access-list

ip dhcp bootp ignore

This command configures the DHCP server to ignore any BOOTP requests it receives. The DHCP server accepts BOOTP requests by default.

The **no** variant of this command configures the DHCP server to accept BOOTP requests. This is the default setting.

Syntax

```
ip dhcp bootp ignore

no ip dhcp bootp ignore
```

Mode Global Configuration

Examples To configure the DHCP server to ignore BOOTP requests, use the command:

```
        awplus# configure terminal
awplus(config)# ip dhcp bootp ignore
```

To configure the DHCP server to respond to BOOTP requests, use the command:

```
        awplus# configure terminal
awplus(config)# no ip dhcp bootp ignore
```

Related Commands show ip dhcp server summary

ip dhcp leasequery enable

Use this command to enable the DHCP server to respond to DHCPLEASEQUERY packets. Enabling the DHCP leasequery feature allows a DHCP relay agent to obtain IP address information directly from the DHCP server using DHCPLEASEQUERY messages.

Use the **no** variant of this command to disable the support of DHCPLEASEQUERY packets.

For more information, see "Enable DHCP Leasequery" on page 60.5.

Syntax

```
ip dhcp leasequery enable
no ip dhcp leasequery enable
```

Default DHCP leasequery support is disabled by default.

Mode Global Configuration

Examples To enable DHCP leasequery support, use the commands:

```
        awplus# configure terminal
awplus(config)# ip dhcp leasequery enable
```

To disable DHCP leasequery support, use the commands:

```
        awplus# configure terminal
awplus(config)# no ip dhcp leasequery enable
```

Related Commands

show counter dhcp-server
show ip dhcp server statistics
show ip dhcp server summary

ip dhcp option

This command creates a user-defined DHCP option. You can then use this option when configuring a DHCP pool, by using the option command. Options with the same number as one of the pre-defined options override the standard option definition. The pre-defined options use the option numbers 1, 3, 6, 15, and 51.

The **no** variant of this command removes either the specified user-defined option, or removes all user-defined options. This also automatically removes the user-defined options from the associated DHCP address pools.

Syntax

```
ip dhcp option <1-254> [name <option-name>] [<option-type>]

no ip dhcp option [<1-254>|<option-name>]
```

Parameter	Description	
<1-254>	The option number of the option. Options with the same number as one of the standard options overrides the standard option definition.	
<option-name>	Option name used to identify the option. You cannot use a number as the option name. Valid characters are any printable character. If the name contains spaces then you must enclose it in "quotation marks". Default: no default	
<option-type>	The option value. You must specify a value that is appropriate to the option type:	
	`asci`	An ASCII text string
	`hex`	A hexadecimal string. Valid characters are the numbers 0–9 and letters a–f. Embedded spaces are not valid. The string must be an even number of characters, from 2 and 256 characters long.
	`ip`	An IPv4 address or mask that has the dotted decimal A.B.C.D notation. To create a list of IP addresses, you must add each IP address individually by using the option command multiple times.
	`integer`	A number from 0 to 4294967295.
	`flag`	A value that either sets (to 1) or unsets (to 0) a flag: **true**, **on**, or **enabled** will set the flag **false**, **off** or **disabled** will unset the flag.

Mode Global Configuration

Examples To define a user-defined ASCII string option as option 66, without a name, use the command:

```
        awplus# configure terminal
awplus(config)# ip dhcp option 66 ascii
```

To define a user-defined hexadecimal string option as option 46, with the name "tcpip-node-type", use the command:

```
        awplus# configure terminal
awplus(config)# ip dhcp option 46 name tcpip-node-type hex
```

To define a user-defined IP address option as option 175, with the name `special-address`, use the command:

```
        awplus# configure terminal
awplus(config)# ip dhcp option 175 name special-address ip
```

To remove the specific user-defined option with the option number `12`, use the command:

```
        awplus# configure terminal
awplus(config)# no ip dhcp option 12
```

To remove the specific user-defined option with the option name `perform-router-discovery`, use the command:

```
        awplus# configure terminal
awplus(config)# no ip dhcp option perform-router-discovery
```

To remove all user-defined option definitions, use the command:

```
        awplus# configure terminal
awplus(config)# no ip dhcp option
```

Related Commands

default-router
dns-server
domain-name
option
service dhcp-server
show ip dhcp server summary
subnet-mask

ip dhcp pool

This command will enter the configuration mode for the pool name specified. If the name specified is not associated with an existing pool, the switch will create a new pool with this name, then enter the configuration mode for the new pool.

Once you have entered the DHCP configuration mode, all commands executed before the next **exit** command will apply to this pool.

You can create multiple DHCP pools on devices with multiple interfaces. This allows the device to act as a DHCP server on multiple interfaces to distribute different information to clients on the different networks.

The **no** variant of this command deletes the specific DHCP pool.

Syntax

```
ip dhcp pool <pool-name>

no ip dhcp pool <pool-name>
```

Parameter	Description
<pool-name>	Description used to identify this DHCP pool. Valid characters are any printable character. If the name contains spaces then you must enclose it in "quotation marks".

Mode Global Configuration

Example To create the DHCP pool called `P2`, use the command:

```
        awplus# configure terminal
awplus(config)# ip dhcp pool P2
```

Related Commands service dhcp-server

ip dhcp-relay agent-option

This command enables the DHCP relay agent to insert the relay agent information option (Option 82) into the client-request packets that it relays to its DHCP server. This allows the relay agent to pass on information to the server about the network location of the client device. The relay agent then strips the Option 82 field out of the server's response, so that the client never sees this field.

When the relay agent appends its Option 82 data into the packet, it first overwrites any pad options present; then if necessary, it increases the packet length to accommodate the option-82 data.

The **no** variant of this command stops the relay agent from appending the Option 82 field onto DHCP requests before forwarding it to the server.

Syntax

```
ip dhcp-relay agent-option
```

```
no ip dhcp-relay agent-option
```

Default The DHCP relay agent feature is disabled by default.

Mode Interface Configuration for a VLAN interface.

Usage Use this command to alter the relay agent's Option 82 setting when your device is the first hop for the DHCP client. To limit the maximum length of the packet, use the ip dhcp-relay max-message-length command.

This command cannot be enabled if DHCP snooping is enabled (service dhcp-snooping command on page 53.26), and vice versa.

Examples To make the relay agent listening on `vlan15` append the Option 82 field, use the commands:

```
          awplus#  configure terminal
  awplus(config)#  interface vlan15
awplus(config-if)# ip dhcp-relay agent-option
```

To stop the relay agent from appending the Option 82 field on `vlan15`, use the commands:

```
          awplus#  configure terminal
  awplus(config)#  interface vlan15
awplus(config-if)# no ip dhcp-relay agent-option
```

Related Commands

ip dhcp-relay agent-option remote-id
ip dhcp-relay information policy
ip dhcp-relay max-message-length
service dhcp-relay

ip dhcp-relay agent-option checking

This command controls the way that the DHCP-relay service deals with packets arriving from the client side that have:

- Option 82 information present in the packet
- a **giaddr** field (relay agent IP address field) of 0.0.0.0

By default such packets are accepted and passed through. This assumes that the Option 82 field has been inserted into the packet by a trusted device, such as a Layer 2 DHCP-snooping switch.

However, if you do not have such a trusted device between the relay switch and the clients, then packets arriving with no relay address but containing Option 82 information are treated with suspicion and dropped.

The command **ip dhcp-relay agent-option checking** will cause such packets to be dropped. Packets which contain Option 82 information, but have a non-zero address in the **giaddr** field will continue to be forwarded.

The **no** variant of this command returns this feature to the default state, whereby the DHCP-relay service does not check the state of the **giaddr** field in packets that contain Option 82 information.

Note The DHCP-relay service might also alter the content of the Option 82 field, if the commands ip dhcp-relay agent-option and ip dhcp-relay information policy have also been configured.

Syntax

```
ip dhcp-relay agent-option checking
```

```
no ip dhcp-relay agent-option checking
```

Mode Interface Configuration for a VLAN interface.

Examples To make the relay agent listening on `vlan10` check the Agent ID sub-option field, use the commands:

```
          awplus# configure terminal
  awplus(config)# interface vlan10
awplus(config-if)# ip dhcp-relay agent-option checking
```

To stop the relay agent on `vlan10` from checking the Agent ID sub-option field, use the commands:

```
          awplus# configure terminal
  awplus(config)# interface vlan10
awplus(config-if)# no ip dhcp-relay agent-option checking
```

Related Commands ip dhcp-relay agent-option remote-id
service dhcp-relay

ip dhcp-relay agent-option remote-id

Use this command to specify the Remote ID sub-option of the Option 82 field the DHCP relay agent inserts into clients' request packets. The Remote ID identifies the device that is inserting the Option 82 information. If a Remote ID is not specified, the Remote ID sub-option is set to the switch's MAC address.

Use the **no** variant of this command to return the Remote ID for an interface.

Syntax

```
ip dhcp-relay agent-option remote-id <remote-id>

no ip dhcp-relay agent-option remote-id
```

Parameter	Description
<remote-id>	An alphanumeric (ASCII) string, 1 to 63 characters in length. Additional characters allowed are hyphen (-), underscore (_) and hash (#). Spaces are not allowed.

Default The Remote ID is set to the switch's MAC address by default.

Mode Interface Configuration for a VLAN interface.

Usage The Remote ID sub-option is included in the DHCP Option 82 field of relayed client DHCP packets if:

- DHCP Option 82 is enabled (ip dhcp-relay agent-option), and
- DHCP relay agent is enabled on the switch (service dhcp-relay)

Examples To set the Remote ID to `myid` for client DHCP packets received on `vlan1`, use the commands:

```
         awplus#  configure terminal
 awplus(config)#  interface vlan1
awplus(config-if)# ip dhcp-relay agent-option remote-id myid
```

To remove the Remote ID specified for `vlan1`, use the commands:

```
         awplus#  configure terminal
 awplus(config)#  interface vlan1
awplus(config-if)# no ip dhcp-relay agent-option remote-id
```

Related Commands

ip dhcp-relay agent-option
ip dhcp-relay agent-option checking
show ip dhcp-relay

ip dhcp-relay information policy

This command sets the policy for how the DHCP relay deals with packets arriving from the client that contain Option 82 information.

If the command **ip dhcp-relay agent-option** has not been configured, then this command has no effect at all - no alteration is made to Option 82 information in packets arriving from the client side.

However, if the command **ip dhcp-relay agent-option** has been configured, this command modifies how the DHCP relay service deals with cases where the packet arriving from the client side already contains Option 82 information.

By default, the relay agent replaces any existing Option 82 field with its own relay agent field. This is equivalent to the functionality of the **replace** parameter.

The **no** variant of this command removes the policy, and returns it to the default behavior - i.e. replacing the existing Option 82 field.

Syntax

```
ip dhcp-relay information policy [append|drop|keep|replace]
```

```
no ip dhcp-relay information policy
```

Parameter	Description
`append`	The relay agent appends the Option 82 field of the packet with its own Option 82 details.
`drop`	The relay agent discards the packet.
`keep`	The relay agent forwards the packet without altering the Option 82 field.
`replace`	The relay agent replaces the existing relay agent details in the Option 82 field with its own details before forwarding the packet.

Mode Interface Configuration for a VLAN interface.

Examples To make the relay agent listening on `vlan15` drop any client requests that already contain Option 82 information, use the command:

```
    awplus(config)# interface vlan15
 awplus(config-if)# ip dhcp-relay information policy drop
```

To remove the DHCP relay information policy set with the **ip dhcp information policy** command, use the command:

```
    awplus(config)# interface vlan15
 awplus(config-if)# no ip dhcp-relay information policy
```

Related Commands ip dhcp-relay agent-option
service dhcp-server

ip dhcp-relay maxhops

This command sets the hop count threshold for discarding BOOTP messages. When the hops field in a BOOTP message exceeds the threshold, the relay agent discards the BOOTP message. The hop count threshold is set to 10 hops by default.

Use the **no** variant of this command negation command to reset the hop count to the default.

Syntax

```
ip dhcp-relay maxhops <1-255>

no ip dhcp-relay maxhops
```

Parameter	Description
<1-255>	The maximum hop count value.

Default The default hop count threshold is 10 hops.

Mode Interface Configuration for a VLAN interface.

Example To set the maximum number of hops to 5 for packets arriving in interface `vlan15`, use the command:

```
    awplus(config)# interface vlan15
 awplus(config-if)# ip dhcp-relay maxhops 5
```

Related Commands service dhcp-relay

ip dhcp-relay max-message-length

This command applies when the switch is acting as a DHCP relay and Option 82 insertion is enabled. It sets the maximum DHCP message length (in bytes) for the DHCP packet with its Option 82 data inserted. From this value it calculates the maximum packet size that it will accept at its input. Packets that arrive greater than this value will be dropped.

The **no** variant of this command sets the maximum message length to its default of 1400 bytes.

Syntax

```
ip dhcp-relay max-message-length <548-1472>
```

```
no ip dhcp-relay max-message-length
```

Parameter	Description
<548-1472>	The maximum DHCP message length (this is the message header plus the inserted DHCP option fields).

Default The default is 1400 bytes.

Mode Interface Configuration for a VLAN interface.

Usage Where a DHCP relay (that has Option 82 insertion enabled) receives a *request* packet from a DHCP client, it will append the Option 82 component data, and forward the packet to the DHCP server. The DHCP client will sometimes issue packets containing pad option fields that can be overwritten with Option 82 data. Where there are insufficient pad option fields to contain all the Option 82 data, the DHCP relay will increase the packet size to accommodate the Option 82 data. If the new (increased) packet size exceeds that defined by the **maximum-message-length** parameter, then the DHCP relay will drop the packet.

Note Before setting this command, you must first run the ip dhcp-relay agent-option command on page 61.16. This will allow the Option 82 fields to be appended.

Example To set the maximum DHCP message length to 1200 for packets arriving in interface `vlan7`, use the command:

```
          awplus# configure terminal
  awplus(config)# interface vlan7
awplus(config-if)# ip dhcp-relay max-message-length 1200
```

To reset the maximum DHCP message length to the default of 1400 for packets arriving in interface `vlan7`, use the command:

```
          awplus# configure terminal
  awplus(config)# interface vlan7
awplus(config-if)# no ip dhcp-relay max-message-length
```

Related Commands service dhcp-relay

ip dhcp-relay server-address

This command adds a DHCP server for the DHCP relay agent to forward client DHCP packets to on a particular interface. You can add up to five DHCP servers on each device interface that the DHCP relay agent is listening on.

The **no** variant of this command deletes the specified DHCP server from the list of servers available to the DHCP relay agent.

For introduction and configuration information about DHCP relay agent see "DHCP Relay Agent Introduction" on page 60.8 and "Configuring the DHCP Relay Agent" on page 60.8.

Syntax

```
ip dhcp-relay server-address {<ipv4-address>|
   <ipv6-address> <server-interface>}

no ip dhcp-relay server-address {<ip-address>|
   <ipv6-address> <server-interface>}
```

Parameter	Description
<ipv4-address>	Specify the IPv4 address of the DHCP server for DHCP relay agent to forward client DHCP packets to, in dotted decimal notation. The IPv4 address uses the format A.B.C.D.
<ipv6-address>	Specify the IPv6 address of the DHCP server for DHCP relay agent to forward client DHCP packets to, in hexadecimal notation.
<server-interface>	Specify the interface name of the DHCP server itself, not the interface of the device as specified in Interface Configuration. The interface name of the DHCP server is only required for a DHCP server with an IPv6 address not with an IPv4 address.

Mode Interface Configuration for a VLAN interface.

Usage For a DHCP server with an IPv6 address you must specify the interface for the DHCP server. See examples below for configuration differences between IPv4 and IPv6 DHCP relay servers.

See also the service dhcp-relay command to enable the DHCP relay agent on your device. The ip dhcp-relay server-address command defines a relay destination on an interface on the device, needed before the DHCP relay agent relays DHCP client packets from a DHCP server.

Examples To enable the DHCP relay agent to relay DHCP packets on interface `vlan2` to the DHCP server with the IPv4 address `192.0.2.200`, use the following command sequence:

```
        awplus# configure terminal
awplus(config)# service dhcp-relay
awplus(config)# interface vlan2
awplus(config-if)# ip dhcp-relay server-address 192.0.2.200
```

To remove the DHCP server with the IPv4 address `192.0.2.200` from the list of servers available to the DHCP relay agent on interface `vlan2`, use the following command sequence:

```
          awplus#  configure terminal
   awplus(config)#  interface vlan2
awplus(config-if)#  no ip dhcp-relay server-address 192.0.2.200
```

To enable the DHCP relay agent on your device to relay DHCP packets on interface `vlan10` to the DHCP server with the IPv6 address `2001:0db8:010d::1` on interface `vlan20`, use the following command sequence:

```
          awplus#  configure terminal
   awplus(config)#  service dhcp-relay
   awplus(config)#  interface vlan10
awplus(config-if)#  ip dhcp-relay server-address
                    2001:0db8:010d::1 vlan20
```

To remove the DHCP server with the IPv6 address `2001:0db8:010d::1` on interface `vlan20` from the list of servers available to the DHCP relay agent on interface `vlan10`, use the following command sequence:

```
          awplus#  configure terminal
   awplus(config)#  interface vlan10
awplus(config-if)#  no ip dhcp-relay server-address
                    2001:0db8:010d::1 vlan20
```

Related Commands service dhcp-relay

lease

This command sets the expiration time for a leased address for the DHCP address pool you are configuring. The time set by the days, hours, minutes and seconds is cumulative. The minimum total lease time that can be configured is 20 seconds. The maximum total lease time that can be configured is 120 days.

Note that if you add a user-defined option 51 using the option command, then you will override any settings created with this command. Option 51 specifies a lease time of 1 day.

Use the **infinite** parameter to set the lease expiry time to infinite (leases never expire).

Use the **no** variant of this command to return the lease expiration time back to the default of one day.

Syntax

```
lease <days> <hours> <minutes> [<seconds>]

lease infinite

no lease
```

Parameter	Description
<days>	The number of days, from 0 to 120, that the lease expiry time is configured for. Default: 1
<hours>	The number of hours, from 0 to 24, that the lease expiry time is configured for. Default: 0
<minutes>	The number of minutes, from 0 to 60, the lease expiry time is configured for. Default: 0
<seconds>	The number of seconds, from 0 to 60, the lease expiry time is configured for.
infinite	The lease never expires.

Default The default lease time is 1 day.

Mode DHCP Configuration

Examples To set the lease expiration time for address pool P2 to 35 minutes, use the commands:

```
            awplus# configure terminal
    awplus(config)# ip dhcp pool P2
awplus(dhcp-config)# lease 0 0 35
```

To set the lease expiration time for the address pool Nerv_Office to 1 day, 5 hours, and 30 minutes, use the commands:

```
            awplus# configure terminal
    awplus(config)# ip dhcp pool Nerv_Office
awplus(dhcp-config)# lease 1 5 30
```

To set the lease expiration time for the address pool P3 to 20 seconds, use the commands:

```
         awplus# configure terminal
 awplus(config)# ip dhcp pool P3
awplus(dhcp-config)# lease 0 0 0 20
```

To set the lease expiration time for the pool to never expire, use the command:

```
awplus(dhcp-config)# lease infinite
```

To return the lease expiration time to the default of one day, use the command:

```
awplus(dhcp-config)# no lease
```

Related Commands option
service dhcp-server

network (DHCP)

This command sets the network (subnet) that the DHCP address pool applies to.

The **no** variant of this command removes the network (subnet) from the DHCP address pool.

Syntax

```
network {<ip-subnet-address/prefix-length>|<ip-subnet-address/mask>}
```

```
no network
```

Parameter	Description
<ip-subnet-address/ prefix-length>	The IPv4 subnet address in dotted decimal notation followed by the prefix length in slash notation.
<ip-subnet-address/ mask>	The IPv4 subnet address in dotted decimal notation followed by the subnet mask in dotted decimal notation.

Mode DHCP Configuration

Usage This command will fail if it would make existing ranges invalid. For example, if they do not lie within the new network you are configuring.

The **no** variant of this command will fail if ranges still exist in the pool. You must remove all ranges in the pool before issuing a **no network** command to remove a network from the pool.

Examples To configure a network for the address pool P2, where the subnet is 192.0.2.5 and the mask is 255.255.255.0, use the commands:

```
          awplus# configure terminal
  awplus(config)# ip dhcp pool P2
awplus(dhcp-config)# network 192.0.2.5/24
```

or you can use dotted decimal notation instead of slash notation for the subnet-mask:

```
          awplus# configure terminal
  awplus(config)# ip dhcp pool P2
awplus(dhcp-config)# network 192.0.2.5 255.255.255.0
```

Related Commands service dhcp-server
subnet-mask

next-server

This command sets the next server address for a DHCP server pool. It is the address of the next server that the client should use in its bootstrap process.

The **no** variant of this command removes the next server address from the DHCP address pool.

Syntax

```
next-server <ip-address>

no next-server
```

Parameter	Description
<ip-address>	The server IP address, entered in dotted decimal notation.

Mode DHCP Configuration

Example To set the next-server address for the address pool P2, use the commands:

```
             awplus# configure terminal
     awplus(config)# ip dhcp pool P2
awplus(dhcp-config)# next-server 192.0.2.2
```

option

This command adds a user-defined option to the DHCP address pool you are configuring. For the **hex**, **integer**, and **flag** option types, if the option already exists, the new option overwrites the existing option's value. Options with an **ip** type can hold a list of IP addresses or masks (i.e. entries that have the A.B.C.D address format), so if the option already exists in the pool, then the new IP address is added to the list of existing IP addresses.

Options with the same number as one of the pre-defined options override the standard option definition. The pre-defined options use the option numbers 1, 3, 6, 15, and 51.

The **no** variant of this command removes the specified user-defined option from the DHCP pool, or all user-defined options from the DHCP pool.

Syntax

```
option [<1-254>|<option-name>] <option-value>

no option [<1-254>|<option-value>]
```

Parameter	Description	
<1-254>	The option number of the option. Options with the same number as one of the standard options overrides the standard option definition.	
<option-name>	Option name associated with the option.	
<option-value>	The option value. You must specify a value that is appropriate to the option type:	
	`hex`	A hexadecimal string. Valid characters are the numbers 0–9 and letters a–f. Embedded spaces are not valid. The string must be an even number of characters, from 2 and 256 characters long.
	`ip`	An IPv4 address or mask that has the dotted decimal A.B.C.D notation. To create a list of IP addresses, you must add each IP address individually using the option command multiple times.
	`integer`	A number from 0 to 4294967295.
	`flag`	A value of either true, on, or enabled to set the flag, or false, off or disabled to unset the flag.

Mode DHCP Configuration

Examples To add the ASCII-type option named `tftp-server-name` to the pool `P2` and give the option the value `server1`, use the commands:

```
           awplus# configure terminal
   awplus(config)# ip dhcp pool P2
awplus(dhcp-config)# option tftp-server-name server1
```

To add the hex-type option named `tcpip-node-type` to the pool `P2` and give the option the value `08af`, use the commands:

```
awplus# configure terminal
awplus(config)# ip dhcp pool P2
awplus(dhcp-config)# option tcpip-node-type 08af
```

To add multiple IP addresses for the ip-type option `175`, use the command:

```
awplus(dhcp-config)# option 175 192.0.2.6
awplus(dhcp-config)# option 175 192.0.2.12
awplus(dhcp-config)# option 175 192.0.2.33
```

To add the option `179` to a pool, and give the option the value `123456`, use the command:

```
awplus(dhcp-config)# option 179 123456
```

To add a user-defined flag option with the name `perform-router-discovery`, use the command:

```
awplus(dhcp-config)# option perform-router-discovery yes
```

To clear all user-defined options from a DHCP address pool, use the command:

```
awplus(dhcp-config)# no option
```

To clear a user-defined option, named `tftp-server-name`, use the command:

```
awplus(dhcp-config)# no option tftp-server-name
```

Related Commands

ip dhcp option
lease
service dhcp-server
show ip dhcp pool

probe enable

Use this command to enable lease probing for a DHCP pool. Probing is used by the DHCP server to check if an IP address it wants to lease to a client is already being used by another host.

The **no** variant of this command disables probing for a DHCP pool.

Syntax

```
probe enable

no probe enable
```

Default Probing is enabled by default.

Mode DHCP Pool Configuration

Examples To enable probing for pool P2, use the commands:

```
          awplus# configure terminal
  awplus(config)# ip dhcp pool P2
awplus(config-if)# probe enable
```

To disable probing for pool P2, use the commands:

```
            awplus# configure terminal
    awplus(config)# ip dhcp pool P2
awplus(dhcp-config)# no probe enable
```

Related Commands

ip dhcp pool
probe packets
probe timeout
probe type
show ip dhcp pool

probe packets

Use this command to specify the number of packets sent for each lease probe. Lease probing is configured on a per-DHCP pool basis. When set to 0 probing is effectively disabled.

The **no** variant of this command sets the number of probe packets sent to the default of 5.

Syntax

```
probe packets <0-10>
```

```
no probe packets
```

Parameter	Description
<0-10>	The number of probe packets sent.

Default The default is 5.

Mode DHCP Pool Configuration

Examples To set the number of probe packets to 2 for pool P2, use the commands:

```
           awplus# configure terminal
   awplus(config)# ip dhcp pool P2
awplus(dhcp-config)# probe packets 2
```

To set the number of probe packets to the default 5 for pool P2, use the commands:

```
           awplus# configure terminal
   awplus(config)# ip dhcp pool P2
awplus(dhcp-config)# no probe packets
```

Related Commands

probe enable
probe timeout
probe type
show ip dhcp pool

probe timeout

Use this command to set the timeout value in milliseconds that the server waits for a response after each probe packet is sent. Lease probing is configured on a per-DHCP pool basis.

The **no** variant of this command sets the probe timeout value to the default setting, 200 milliseconds.

Syntax

```
probe timeout <50-5000>
```

```
no probe timeout
```

Parameter	Description
<50-5000>	Timeout interval in milliseconds.

Default The default timeout interval is 200 milliseconds.

Mode DHCP Pool Configuration

Examples To set the probe timeout value to 500 milliseconds for pool P2, use the commands:

```
              awplus# configure terminal
      awplus(config)# ip dhcp pool P2
 awplus(dhcp-config)# probe timeout 500
```

To set the probe timeout value for pool P2 to the default, 200 milliseconds, use the commands:

```
              awplus# configure terminal
      awplus(config)# ip dhcp pool P2
 awplus(dhcp-config)# no probe timeout
```

Related Commands

probe enable
probe packets
probe type
show ip dhcp pool

probe type

Use this command to set the probe type for a DHCP pool. The probe type specifies how the DHCP server checks whether an IP address is being used by other hosts, referred to as lease probing. If **arp** is specified, the server sends an ARP request to determine if an address is in use. If **ping** is specified, the server will send an ICMP Echo Request (ping).

The **no** variant of this command sets the probe type to the default setting, ping.

Syntax

```
probe type {arp|ping}
```

```
no probe type
```

Parameter	Description
`arp`	Probe using ARP.
`ping`	Probe using ping.

Default The default probe type is ping.

Mode DHCP Pool Configuration

Examples To set the probe type to `arp` for the pool `P2`, use the commands:

```
             awplus# configure terminal
     awplus(config)# ip dhcp pool P2
awplus(dhcp-config)# probe type arp
```

To set the probe type for the pool `P2` to the default, `ping`, use the commands:

```
             awplus# configure terminal
     awplus(config)# ip dhcp pool P2
awplus(dhcp-config)# no probe type
```

Related Commands

ip dhcp pooll
probe enable
probe packets
probe timeout
show ip dhcp pool

range

This command adds an address range to the DHCP address pool you are configuring. The DHCP server responds to client requests received from the pool's network. It assigns an IP addresses within the specified range. The IP address range must lie within the network. You can add multiple address ranges and individual IP addresses for a DHCP pool by using this command multiple times.

The **no** variant of this command removes an address range from the DHCP pool. Use the **no range all** command to remove all address ranges from the DHCP pool.

Syntax

```
range <ip-address> [<ip-address>]

no range <ip-address> [<ip-address>]

no range all
```

Parameter	Description
<ip-address>	IPv4 address range for DHCP clients, in dotted decimal notation. The first IP address is the low end of the range, the second IP address is the high end. Specify only one IP address to add an individual IP address to the address pool.

Mode DHCP Configuration

Examples To add an address range of `192.0.2.5` to `192.0.2.16` to the pool `Nerv_Office`, use the command:

```
           awplus# configure terminal
   awplus(config)# ip dhcp pool Nerv_Office
awplus(dhcp-config)# range 192.0.2.5 192.0.2.16
```

To add the individual IP address `192.0.2.2` to a pool, use the command:

```
awplus(dhcp-config)# range 192.0.2.2
```

To remove all address ranges from a pool, use the command:

```
awplus(dhcp-config)# no range all
```

Related Commands

ip dhcp pool
service dhcp-server
show ip dhcp pool

service dhcp-relay

This command enables the DHCP relay agent on the device. However, on a given IP interface, no DHCP forwarding takes place until at least one DHCP server is specified to forward/relay all clients' DHCP packets to.

The **no** variant of this command disables the DHCP relay agent on the device for all interfaces.

Syntax

```
service dhcp-relay

no service dhcp-relay
```

Mode Global Configuration

Usage A maximum number of 400 DHCP relay agents (one per interface) can be configured on the device. Once this limit has been reached, any further attempts to configure DHCP relay agents will not be successful.

Default The DHCP-relay service is enabled by default.

Examples To enable the DHCP relay global function, use the command:

```
        awplus# configure terminal
awplus(config)# service dhcp-relay
```

To disable the DHCP relay global function, use the command:

```
        awplus# configure terminal
awplus(config)# no service dhcp-relay
```

Related Commands

ip dhcp-relay agent-option
ip dhcp-relay agent-option checking
ip dhcp-relay information policy
ip dhcp-relay maxhops
ip dhcp-relay server-address

service dhcp-server

This command enables the DHCP server on your device. The server then listens for DHCP requests on all IP interfaces. It will not run if there are no IP interfaces configured.

The **no** variant of this command disables the DHCP server.

Syntax

```
service dhcp-server

no service dhcp-server
```

Mode Global Configuration

Example To enable the DHCP server, use the command:

```
        awplus# configure terminal
awplus(config)# service dhcp-server
```

Related Commands

ip dhcp pool
show ip dhcp server summary
subnet-mask

show counter dhcp-client

This command shows counters for the DHCP client on your device.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax

```
show counter dhcp-client
```

Mode User Exec and Privileged Exec

Example To display the message counters for the DHCP client on your device, use the command:

```
awplus# show counter dhcp-client
```

Output Figure 61-1: Example output from the **show counter dhcp-client** command

```
show counter dhcp-client

DHCPDISCOVER out      ......... 10
DHCPREQUEST out       ......... 34
DHCPDECLINE out       ......... 4
DHCPRELEASE out       ......... 0
DHCPOFFER in          ......... 22
DHCPACK in            ......... 18
DHCPNAK in            ......... 0
```

Table 61-1: Parameters in the output of the **show counter dhcp-client** command

Parameter	Description
`DHCPDISCOVER out`	The number of DHCP Discover messages sent by the client.
`DHCPREQUEST out`	The number of DHCP Request messages sent by the client.
`DHCPDECLINE out`	The number of DHCP Decline messages sent by the client.
`DHCPRELEASE out`	The number of DHCP Release messages sent by the client.
`DHCPOFFER in`	The number of DHCP Offer messages received by the client.
`DHCPACK in`	The number of DHCP Acknowledgement messages received by the client.
`DHCPNAK in`	The number of DHCP Negative Acknowledgement messages received by the client.

Related Commands ip address dhcp

show counter dhcp-relay

This command shows counters for the DHCP relay agent on your device.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax

```
show counter dhcp-relay
```

Mode

User Exec and Privileged Exec

Example

To display counters for the DHCP relay agent on your device, use the command:

```
awplus# show counter dhcp-relay
```

Output

Figure 61-2: Example output from the **show counter dhcp-relay** command

```
show counter dhcp-relay

Requests In             ......... 4
Replies In              ......... 4
Relayed To Server       ......... 4
Relayed To Client       ......... 4
Out To Server Failed    ......... 0
Out To Client Failed    ......... 0
Invalid hlen            ......... 0
Bogus giaddr            ......... 0
Corrupt Agent Option    ......... 0
Missing Agent Option    ......... 0
Bad Circuit ID          ......... 0
Missing Circuit ID      ......... 0
Option Insert Failed    ......... 0
```

Table 61-2: Parameters in the output of the **show counter dhcp-relay** command

Parameter	Description
`Requests In`	The number of DHCP Request messages received from clients.
`Replies In`	The number of DHCP Reply messages received from servers.
`Relayed To Server`	The number of DHCP Request messages relayed to servers.
`Relayed To Client`	The number of DHCP Reply messages relayed to clients.
`Out To Server Failed`	The number of failures when attempting to send request messages to servers. This is an internal debugging counter.
`Out To Client Failed`	The number of failures when attempting to send reply messages to clients. This is an internal debugging counter.
`Invalid hlen`	The number of incoming messages dropped due to an invalid hlen field.
`Bogus giaddr`	The number of incoming DHCP Reply messages dropped due to bogus giaddr field.
`Corrupt Agent Option`	The number of incoming DHCP Reply messages dropped due to corrupt agent option.
`Missing Agent Option`	The number of incoming DHCP Reply messages dropped due to missing agent option.

Table 61-2: Parameters in the output of the **show counter dhcp-relay** command(cont.)

Parameter	Description
`Bad Circuit ID`	The number of incoming DHCP Reply messages dropped due to bad circuit ID.
`Missing Circuit ID`	The number of incoming DHCP Reply messages dropped due to missing circuit ID.
`Option Insert Failed`	The number of incoming DHCP Request messages dropped due to an error adding the relay agent information (option-82). This counter increments when: the relay agent is set to drop packets with the Option 82 field already filled by another relay agent. This policy is set with the ip dhcp-relay information policy command. there is a packet error that stops the relay agent from being able to append the packet with its relay agent option information.

Related Commands

service dhcp-relay
show ip dhcp-relay

show counter dhcp-server

This command shows counters for the DHCP server on your device.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax `show counter dhcp-server`

Mode User Exec and Privileged Exec

Example To display counters for the DHCP server on your device, use the command:

```
awplus# show counter dhcp-server
```

Output Figure 61-3: Example output from the **show counter dhcp-server** command

```
DHCP server counters
DHCPDISCOVER in        ......... 20
DHCPREQUEST in         ......... 12
DHCPDECLINE in         ......... 1
DHCPRELEASE in         ......... 0
DHCPINFORM in          ......... 0
DHCPOFFER out          ......... 8
DHCPACK out            ......... 4
DHCPNAK out            ......... 0
BOOTREQUEST in         ......... 0
BOOTREPLY out          ......... 0
DHCPLEASEQUERY in        ....... 0
DHCPLEASEUNKNOWN out     ....... 0
DHCPLEASEACTIVE out      ....... 0
DHCPLEASEUNASSIGNED out ....... 0
```

Table 61-3: Parameters in the output of the **show counter dhcp-server** command

Parameter	Description
DHCPDISCOVER in	The number of Discover messages received by the DHCP server.
DHCPREQUEST in	The number of Request messages received by the DHCP server.
DHCPDECLINE in	The number of Decline messages received by the DHCP server.
DHCPRELEASE in	The number of Release messages received by the DHCP server.
DHCPINFORM in	The number of Inform messages received by the DHCP server.
DHCPOFFER out	The number of Offer messages sent by the DHCP server.
DHCPACK out	The number of Acknowledgement messages sent by the DHCP server.
DHCPNAK out	The number of Negative Acknowledgement messages sent by the DHCP server. The server sends these after receiving a request that it cannot fulfil because either there are no available IP addresses in the related address pool, or the request has come from a client that doesn't fit the network setting for an address pool.

Table 61-3: Parameters in the output of the **show counter dhcp-server** command(cont.)

Parameter	Description
`BOOTREQUEST in`	The number of bootp messages received by the DHCP server from bootp clients.
`BOOTREPLY out`	The number of bootp messages sent by the DHCP server to bootp clients.
`DHCPLEASEQUERY in`	The number of Lease Query messages received by the DHCP server from DHCP relay agents.
`DHCPLEASEUNKNOWN out`	The number of Lease Unknown messages sent by the DHCP server to DHCP relay agents.
`DHCPLEASEACTIVE out`	The number of Lease Active messages sent by the DHCP server to DHCP relay agents.
`DHCPLEASEUNASSIGNED out`	The number of Lease Unassigned messages sent by the DHCP server to DHCP relay agents.

Related Commands

service dhcp-server
show ip dhcp binding
show ip dhcp server statistics
show ip dhcp pool

show dhcp lease

This command shows details about the leases that the DHCP client has acquired from a DHCP server for interfaces on the device.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax

```
show dhcp lease [<interface>]
```

Parameter	Description
<interface>	Interface name to display DHCP lease details for.

Mode User Exec and Privileged Exec

Example To show the current lease expiry times for all interfaces, use the command:

```
awplus# show dhcp lease
```

To show the current lease for vlan1, use the command:

```
awplus# show dhcp lease vlan1
```

Output Figure 61-4: Example output from the **show dhcp lease** command

```
Interface vlan1
----------------------------------------------------------------
IP Address:                  192.168.22.4
Expires:                     13 Mar 2007 20:10:19
Renew:                       13 Mar 2007 18:37:06
Rebind:                      13 Mar 2007 19:49:29
Server:
Options:
  subnet-mask                255.255.255.0
  routers                    19.18.2.100,12.16.2.17
  dhcp-lease-time            3600
  dhcp-message-type          5
  domain-name-servers        192.168.100.50,19.88.200.33
  dhcp-server-identifier     192.168.22.1
  domain-name                alliedtelesis.com

Interface vlan2
----------------------------------------------------------------
IP Address:                  100.8.16.4
Expires:                     13 Mar 2007 20:15:39
Renew:                       13 Mar 2007 18:42:25
Rebind:                      13 Mar 2007 19:54:46
Server:
Options:
  subnet-mask                255.255.0.0
  routers                    10.58.1.51
  dhcp-lease-time            1000
  dhcp-message-type          5
  dhcp-server-identifier     100.8.16.1
```

Related Commands ip address dhcp

show ip dhcp binding

This command shows the lease bindings that the DHCP server has allocated clients.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax

```
show ip dhcp binding [<ip-address>|<address-pool>]
```

Parameter	Description
<ip-address>	IPv4 address of a leased IP address, in dotted decimal notation. This displays the lease information for the specified IP address.
<address-pool>	Name of an address pool. This displays the lease information for all clients within the address pool.

Mode User Exec and Privileged Exec

Examples To display all leases for every client in all address pools, use the command:

```
awplus# show ip dhcp binding
```

To display the details for the leased IP address `172.16.2.16`, use the command:

```
awplus# show ip dhcp binding 172.16.2.16
```

To display the leases from the address pool `MyPool`, use the command:

```
awplus# show ip dhcp binding MyPool
```

Output Figure 61-5: Example output from the **show ip dhcp binding** command

```
Pool 30_2_network Network 172.16.2.0/24
DHCP Client Entries
IP Address      ClientId                    Type           Expiry
-------------------------------------------------------------------------------
172.16.2.100    0050.fc82.9ede              Dynamic        21 Sep 2007 19:02:58
172.16.2.101    000e.a6ae.7c14              Static         Infinite
172.16.2.102    000e.a6ae.7c4c              Static         Infinite
172.16.2.103    000e.a69a.ac91              Static         Infinite
172.16.2.104    00e0.189d.5e41              Static         Infinite
172.16.2.150    00e0.2b04.5800              Static         Infinite
172.16.2.167    4444.4400.35c3              Dynamic        21 Sep 2007 14:58:41
```

Related Commands

clear ip dhcp binding
ip dhcp pool
lease
range
service dhcp-server
show ip dhcp pool

show ip dhcp pool

This command displays the configuration details and system usage of the DHCP address pools configured on the device.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax `show ip dhcp pool [<address-pool>]`

Parameter	Description
<address-pool>	Name of a specific address pool. This displays the configuration of the specified address pool only.

Mode User Exec and Privileged Exec

Example

```
awplus# show ip dhcp pool
```

Output Figure 61-6: Example output from the **show ip dhcp pool** command

```
Pool p1 :
  network: 192.168.1.0/24
  address ranges:
    addr: 192.168.1.10 to 192.168.1.18
  static host addresses:
    addr: 192.168.1.12     MAC addr: 1111.2222.3333
  lease <days:hours:minutes:seconds> <1:0:0:0>
  subnet mask: 255.255.255.0 (pool's network mask)
  Probe:                       Default Values
    Status:       Enabled      [Enabled]
    Type:         ARP          [Ping]
    Packets:      2            [5]
    Timeout:      200 msecs    [200]
  Dynamic addresses:
    Total:        8
    Leased:       2
    Utilization:  25.0 %
  Static host addresses:
    Total:        1
    Leased:       1
```

Output Figure 61-7: Example output from the **show ip dhcp pool** command with IP address 192.168.1.12 assigned to a VLAN interface on the device:

```
Pool p1 :
  network: 192.168.1.0/24
  address ranges:
    addr: 192.168.1.10 to 192.168.1.18
            (interface addr 192.168.1.12 excluded)
            (static host addr 192.168.1.12 excluded)
  static host addresses:
    addr: 192.168.1.12     MAC addr: 1111.2222.3333
            (= interface addr, so excluded)
  lease <days:hours:minutes:seconds> <1:0:0:0>
  subnet mask: 255.255.255.0 (pool's network mask)
  Probe:                        Default Values
    Status:       Enabled       [Enabled]
    Type:         ARP           [Ping]
    Packets:      2             [5]
    Timeout:      200 msecs     [200]
  Dynamic addresses:
    Total:        8
    Leased:       2
    Utilization:  25.0 %
  Static host addresses:
    Total:        1
    Leased:       1
```

Table 61-4: Parameters in the output of the **show ip dhcp pool** command

Parameter	Description
`Pool`	Name of the pool.
`network`	Subnet and mask length of the pool.
`address ranges`	Individual IP addresses and address ranges configured for the pool. The DHCP server can offer clients an IP address from within the specified ranges only. Any of these addresses that match an interface address on the device, or a static host address configured in the pool, will be automatically excluded from the range, and a message to this effect will appear beneath the range entry.
`static host addresses`	The static host addresses configured on the pool. Each IP address is permanently assigned to the client with the matching MAC address. Any of these addresses that match an interface address on the device will be automatically excluded, and a message to this effect will appear beneath the static host entry.
`lease <days:hours:minutes>`	The lease duration for address allocated by this pool.
`domain`	The domain name sent by the pool to clients. This is the domain name that the client should use when resolving host names using DNS.
`subnet mask`	The subnet mask sent by the pool to clients.
`Probe - Status`	Whether lease probing is enabled or disabled.
`Probe - Type`	The lease probe type configured. Either ping or ARP.

Table 61-4: Parameters in the output of the **show ip dhcp pool** command(cont.)

Parameter	Description
`Probe - Packets`	The number of packets sent for each lease probe in the range 0 to 10.
`Probe - Timeout`	The timeout value in milliseconds to wait for a response after each probe packet is sent. In the range 50 to 5000.
`dns servers`	The DNS server addresses sent to by the pool to clients.
`default-router(s)`	The default router addresses sent by the pool to clients.
`user-defined options`	The list of user-defined options sent by the pool to clients.
`Dynamic addresses - Total`	The total number of IP addresses that have been configured in the pool for dynamic allocation to DHCP clients.
`Dynamic addresses - Leased`	The number of IP addresses in the pool that have been dynamically allocated (leased) to DHCP clients.
`Dynamic addresses - Utilization`	The percentage of IP addresses in the pool that are currently dynamically allocated to clients.
`Static host addresses - Total`	The number of static IP addresses configured in the pool for specific DHCP client hosts.
`Static host addresses - Leased`	The number of static IP addresses assigned to specific DHCP client hosts.

Related Commands

ip dhcp pool
probe enable
probe packets
probe timeout
probe type
range
service dhcp-server
subnet-mask

show ip dhcp-relay

This command shows the configuration of the DHCP relay agent on each interface.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax

```
show ip dhcp-relay [interface <interface-name>]
```

Parameter	Description
<interface-name>	Name of a specific interface. This displays the DHCP configuration for the specified interface only.

Mode User Exec and Privileged Exec

Example To display the DHCP relay agent's configuration on the interface `vlan100`, use the command:

```
awplus# show ip dhcp-relay interface vlan100
```

Output Figure 61-8: Example output from the **show ip dhcp-relay** command

```
DHCP Relay Service is enabled

vlan100 is up, line protocol is up
Maximum hop count is 10
Insertion of Relay Agent Option is disabled
Checking of Relay Agent Option is disabled
The Remote Id string for Relay Agent Option is 0000.cd28.074c
Relay information policy is to append new relay agent
information
List of servers :   192.168.1.200
```

Related Commands

ip dhcp-relay agent-option
ip dhcp-relay agent-option checking
ip dhcp-relay information policy
ip dhcp-relay maxhops
ip dhcp-relay server-address

show ip dhcp server statistics

This command shows statistics related to the DHCP server.

You can display the server counters using the show counter dhcp-server command as well as with this command.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax

```
show ip dhcp server statistics
```

Mode User Exec and Privileged Exec

Example To display the server statistics, use the command:

```
awplus# show ip dhcp server statistics
```

Output

Figure 61-9: Example output from the **show counter dhcp server statistics** command

```
DHCP server counters
DHCPDISCOVER in       ......... 20
DHCPREQUEST in        ......... 12
DHCPDECLINE in        ......... 1
DHCPRELEASE in        ......... 0
DHCPINFORM in         ......... 0
DHCPOFFER out         ......... 8
DHCPACK out           ......... 4
DHCPNAK out           ......... 0
BOOTREQUEST in        ......... 0
BOOTREPLY out         ......... 0
DHCPLEASEQUERY in       ....... 0
DHCPLEASEUNKNOWN out    ....... 0
DHCPLEASEACTIVE out     ....... 0
DHCPLEASEUNASSIGNED out ....... 0
```

Figure 61-10: Parameters in the output of the **show counter dhcp server statistics** command

Parameter	Description
`DHCPDISCOVER in`	The number of Discover messages received by the DHCP server.
`DHCPREQUEST in`	The number of Request messages received by the DHCP server.
`DHCPDECLINE in`	The number of Decline messages received by the DHCP server.
`DHCPRELEASE in`	The number of Release messages received by the DHCP server.
`DHCPINFORM in`	The number of Inform messages received by the DHCP server.
`DHCPOFFER out`	The number of Offer messages sent by the DHCP server.
`DHCPACK out`	The number of Acknowledgement messages sent by the DHCP server.

Figure 61-10: Parameters in the output of the **show counter dhcp server statistics** command(cont.)

`DHCPNAK out`	The number of Negative Acknowledgement messages sent by the DHCP server. The server sends these after receiving a request that it cannot fulfil because either there are no available IP addresses in the related address pool, or the request has come from a client that doesn't fit the network setting for an address pool.
`BOOTREQUEST in`	The number of bootp messages received by the DHCP server from bootp clients.
`BOOTREPLY out`	The number of bootp messages sent by the DHCP server to bootp clients.
`DHCPLEASEQUERY in`	The number of Lease Query messages received by the DHCP server from DHCP relay agents.
`DHCPLEASEUNKNOWN out`	The number of Lease Unknown messages sent by the DHCP server to DHCP relay agents.
`DHCPLEASEACTIVE out`	The number of Lease Active messages sent by the DHCP server to DHCP relay agents.
`DHCPLEASEUNASSIGNED out`	The number of Lease Unassigned messages sent by the DHCP server to DHCP relay agents.

Related Commands

show counter dhcp-server
service dhcp-server
show ip dhcp binding
show ip dhcp pool

show ip dhcp server summary

This command shows the current configuration of the DHCP server. This includes:

- whether the DHCP server is enabled
- whether the DHCP server is configured to ignore BOOTP requests
- whether the DHCP server is configured to support DHCP lease queries
- the details of any user-defined options
- a list of the names of all DHCP address pools currently configured

This show command does not include any configuration details of the address pools. You can display these using the show ip dhcp pool command.

For information on output options, see "Controlling "show" Command Output" on page 1.41.

Syntax

```
show ip dhcp server summary
```

Mode

User Exec and Privileged Exec

Example

To display the current configuration of the DHCP server, use the command:

```
awplus# show ip dhcp server summary
```

Output

Figure 61-11: Example output from the **show ip dhcp server summary** command

```
DHCP Server service is disabled
BOOTP ignore is disabled
DHCP leasequery support is disabled
Pool list: p2
```

Related Commands

ip dhcp leasequery enable
ip dhcp pool
service dhcp-server

subnet-mask

This command sets the subnet mask option for a DHCP address pool you are configuring. Use this command to specify the client's subnet mask as defined in RFC 950. This sets the subnet details using the pre-defined option 1. Note that if you create a user-defined option 1 using the option command, then you will override any settings created with this command. If you do not specify a subnet mask using this command, then the pool's network mask (specified using the next-server command) is applied.

The **no** variant of this command removes a subnet mask option from a DHCP pool. The pool reverts to using the pool's network mask.

Syntax

```
subnet-mask <mask>
```

```
no subnet-mask
```

Parameter	Description
<mask>	Valid IPv4 subnet mask, in dotted decimal notation.

Mode DHCP Configuration

Examples To set the subnet mask option to 255.255.255.0 for DHCP pool P2, use the commands:

```
            awplus# configure terminal
    awplus(config)# ip dhcp pool P2
awplus(dhcp-config)# subnet-mask 255.255.255.0
```

To remove the subnet mask option from DHCP pool P2, use the commands:

```
            awplus# configure terminal
    awplus(config)# ip dhcp pool P2
awplus(dhcp-config)# no subnet-mask
```

Related Commands

default-router
dns-server
domain-name
next-server
option
service dhcp-server
show ip dhcp pool

Chapter 62: SNMP Introduction

Introduction

The Simple Network Management Protocol (SNMP) is the network management protocol of choice for the Internet and IP-based internetworks.

This chapter describes the main features of SNMP Version 1 (SNMPv1), SNMP Version 2c (SNMPv2c) and Version 3 (SNMPv3). It also describes support for SNMP on the switch, and how to configure the switch's SNMP agent.

Unless a particular version of SNMP is named, "SNMP" in this chapter refers to versions SNMPv1, SNMPv2c and SNMPv3.

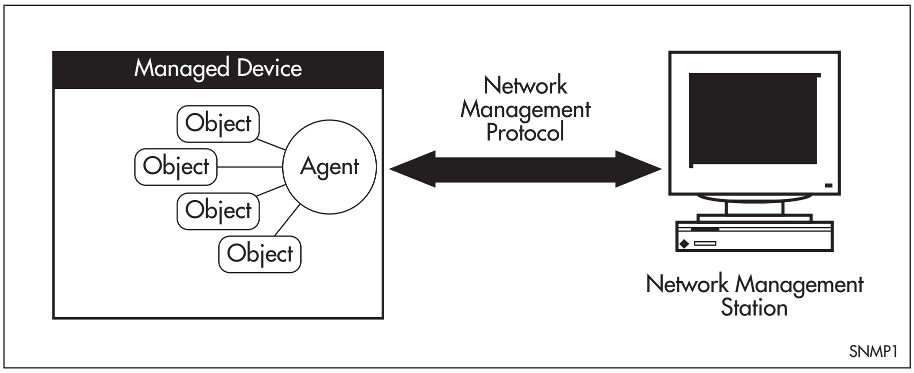
See also Chapter 63, SNMP Commands and Chapter 64, SNMP MIBs.

Network Management Framework

A network management system has the following components:

- One or more **managed devices**, each containing an agent that provides the management functions. A managed device may be any computing device with a network capability, for example, a host system, workstation, terminal server, printer, router, switch, bridge, hub or repeater.
- One or more **Network Management Stations (NMS)**. An NMS is a host system running a network management protocol and network management applications, enabling the user to manage the network.
- A **network management protocol** used by the NMS and agents to exchange information.

Figure 62-1: Components of a network management system



The Internet-standard Network Management Framework is the framework used for network management in the Internet. The framework was originally defined by the following documents:

- RFC 1155, *Structure and identification of management information for TCP/IP based internets* (referred to as the SMI), details the mechanisms used to describe and name the objects to be managed.
- RFC 1213, *Management Information Base for network management of TCP/ IP-based internets: MIB-II* (referred to as MIB-II), defines the core set of managed objects for the Internet suite of protocols. The set of managed objects can be extended by adding other MIBs specific to particular protocols, interfaces or network devices.
- RFC 1157, *A Simple Network Management Protocol (SNMP)*, is the protocol used for communication between management stations and managed devices.

Subsequent documents that have defined SNMPv2c are:

- RFC 1901, *Introduction to Community-based SNMPv2*
- RFC 1902, *Structure of Management Information for Version 2 of the Simple Network Management Protocol (SNMPv2)*
- RFC 1903, *Textual Conventions for Version 2 of the Simple Network Management Protocol (SNMPv2)*
- RFC 1904, *Conformance Statements for Version 2 of the Simple Network Management Protocol*
- RFC 1905, *Protocol Operations for Version 2 of the Simple Network Management Protocol (SNMPv2)*
- RFC 1906, *Transport Mappings for Version 2 of the Simple Network Management Protocol (SNMPv2)*
- RFC 1907, *Management Information Base for Version 2 of the Simple Network Management Protocol (SNMPv2)*
- RFC 2576, *Coexistence between Version 1, Version 2, and Version 3 of the Internet-standard Network Management Framework*
- RFC 2578, *Structure of Management Information Version 2 (SMIv2)*
- RFC 2579, *Textual Conventions for SMIv2*
- RFC 2580, *Conformance Statements for SMIv2*

Subsequent documents that have defined SNMPv3 are:

- RFC 3410, *Introduction and Applicability Statements for Internet Standard Management Framework*
- RFC 3411, *An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks*
- RFC 3412, *Message Processing and Dispatching for the Simple Network Management Protocol (SNMP)*
- RFC 3413, *Simple Network Management Protocol (SNMP) Applications*
- RFC 3414, *User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3)*
- RFC 3415, *View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)*
- RFC 3416, *Version 2 of the Protocol Operations for the Simple Network Management Protocol (SNMP)*
- RFC 3417, *Transport Mappings for the Simple Network Management Protocol (SNMP)*
- RFC 3418, *Management Information Base (MIB) for the Simple Network Management Protocol (SNMP)*

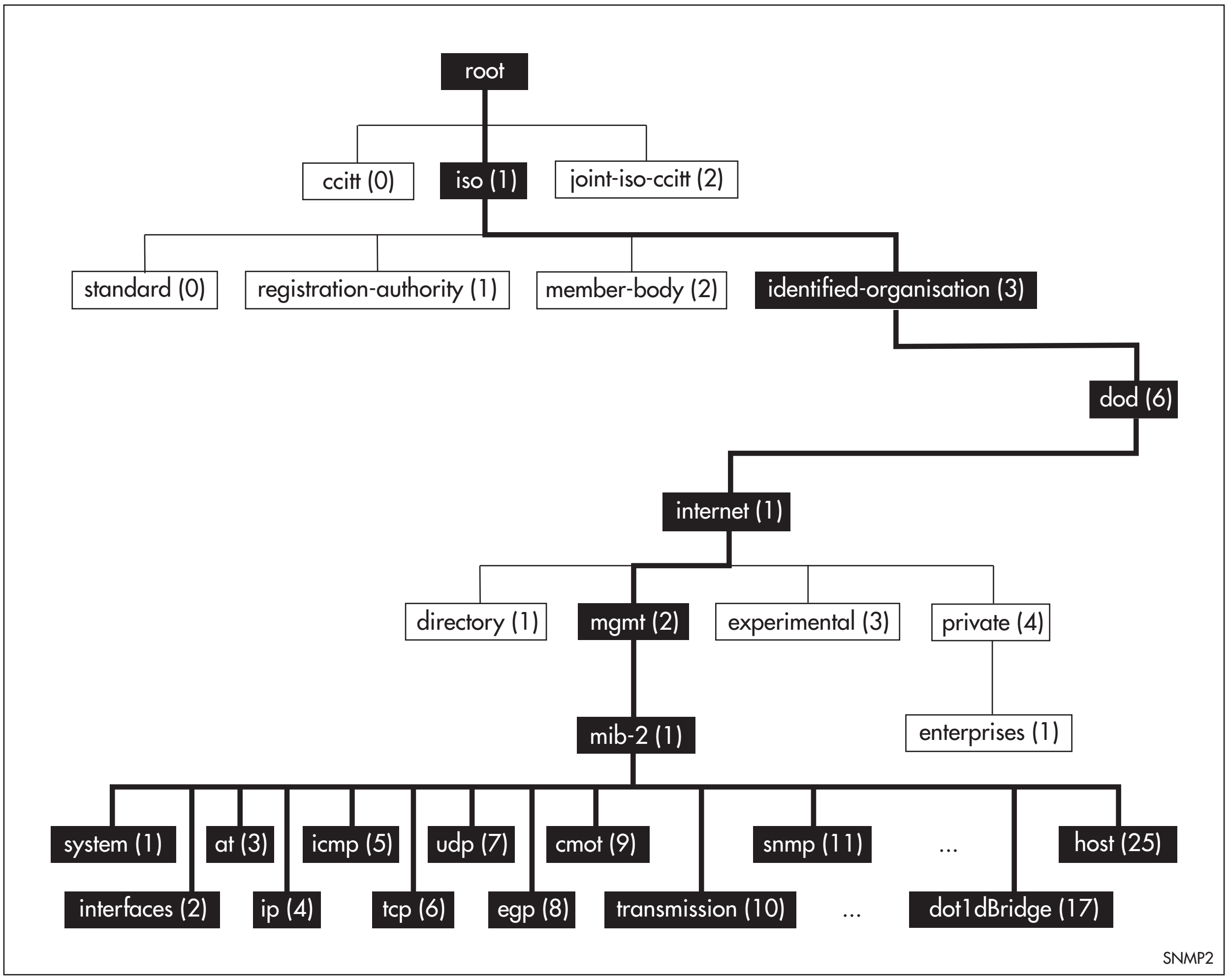
Structure of Management Information

The structure of management information (SMI) defines the schema for a collection of managed objects residing in a virtual store called the management information base (MIB). The information in a MIB includes administrative and operational configuration information, as well as counters of system events and activities.

The MIB is organized into a tree-like hierarchy in which nodes are each assigned an identifier consisting of a non-negative integer and an optional brief textual description.

Each managed object is represented by a leaf node and is defined by its name, syntax, access mode, status and description. It can also be specifically identified by its unique position within the tree. This position is expressed as a series of dot-delimited sub-identifiers that start at the root node and end in the sub-identifier at the particular object's leaf node. For example, in Figure 62-2 the object named interfaces would be uniquely identified by the string of individual sub-identifiers, 1.3.6.1.2.1.2.

Figure 62-2: Top levels of the Internet-standard Management Information Base (MIB)



Objects defined in the Internet-standard MIB (MIB-II) reside in the mib(1) sub-tree.

Names

Names are used to identify managed objects, and are hierarchical in nature. An object identifier is a globally unique, authoritatively assigned sequence of non-negative integers which traverse the MIB tree from the root to the node containing the object.

Object identifiers may be represented in one of the following forms:

- Dotted notation lists the integer values found by traversing the tree from the root to the node in question, separated by dots. For example, the following identifies the MIB-II sub-tree:

  ```
  1.3.6.1.2.1
  ```

  The following identifies the sysDescr object in the system group of MIB-II:

  ```
  1.3.6.1.2.1.1.1
  ```

- Textual notation lists the textual descriptions found by traversing the tree from the root to the node in question, separated by spaces and enclosed in braces. For following example identifies the internet sub-tree:

  ```
  { iso org dod 1}
  ```

  The name may be abbreviated to a relative form. The following example identifies the first (directory) node of the internet sub-tree:

  ```
  { internet 1}
  ```

- Combined notation lists both the integer values and textual descriptions found by traversing the tree from the root to the node in question. The integer value is placed in parentheses after the textual description. The labels are separated by spaces and enclosed in braces. For example, the following identifies the first (directory) node in the internet sub-tree:

  ```
  {iso(1) org(3) dod(6) internet(1) 1}
  ```

  The name may be abbreviated to the following:

  ```
  directory(1)
  ```

Since there is no effective limit to the magnitude of non-negative integers, and no effective limit to the depth of the tree, the MIB provides an unlimited name space.

An object is also usually assigned an object descriptor. The object descriptor is a unique, mnemonic, printable string intended for humans to use when discussing the MIB.

Instances

Objects are just templates for data types. An actual value that can be manipulated by an NMS is an instance of an object. An instance is named by appending an instance identifier to the end of the object's object identifier. The instance identifier depends on the object's data type:

- If the object is not a column in a table, the instance identifier is 0 (zero). For example, the instance of the sysDescr object is:

  ```
  sysDescr.0
  or 1.3.6.1.2.1.1.1.0
  ```

- If the object is a column in a table, the method used to assign an instance identifier varies. Typically, the value of the index column or columns is used.

The object ifTable in MIB-II contains information about interfaces and is indexed by the interface number, ifIndex. The instance of the ifDescr object for the first interface is:

```
ifDescr.1
or 1.3.6.1.2.1.2.2.1.2.1
```

If the index column is an IP address, the entire IP address is used as the instance identifier. The object ipRouteTable in MIB-II contains information about IP routes and is indexed by the destination address, ipRouteDest. The instance of the ipRouteNextHop object for the route 131.203.9.0 is:

```
ipRouteNextHop.131.203.9.0
or 1.3.6.1.2.1.4.21.1.7.131.203.9.0
```

If the table has more than one index, the values of all the index columns are combined to form the instance identifier. The object tcpConnTable in MIB-II contains information about existing TCP connections and is indexed by the local IP address (tcpConnLocalAddress), the local port number (tcpConnLocalPort), the remote IP address (tcpConnRemAddress) and the remote port number (tcpConnRemPort) of the TCP connection. The instance of the tcpConnState object for the connection between 131.203.8.36,23 and 131.203.9.197,1066 is:

tcpConnState.131.203.8.36.23.131.203.9.197.1066
or 1.3.6.1.2.1.6.13.1.1.131.203.8.36.23.131.203.9.197.1066

Syntax

The syntax of an object describes the abstract data structure corresponding to that object type. For example, INTEGER or OCTET STRING.

Access

The access mode of an object describes the level of access for the object.

Access modes for MIB objects:

Access	Description
Read-only	The object's value can be read but not set.
Read-write	The object's value can be read and set.
Write-only	The object's value can be set but not read.
Not-accessible	The object's value cannot be read or set.

Status

The status of an object describes the implementation requirements for the object.

Status values for MIB objects:

Status	Description
Mandatory	Managed devices must implement the object.
Optional	Managed devices may implement the object.
Obsolete	Managed devices need no longer implement the object.
Deprecated	Managed devices should implement the object. However, the object may be deleted from the next version of the MIB. A new object with equal or superior functionality is defined.

Description

The definition of an object may include an optional textual description of the meaning and use of the object. This description is often essential for successful understanding of the object.

The SNMP Protocol

The SNMP protocol provides a mechanism for management entities, or stations, to extract information from the Management Information Base (MIB) of a managed device.

The normal method of accessing information in a MIB is to use a Network Management Station (NMS), typically a PC or workstation, to send commands to the managed device (in this case the switch) using the SNMP protocol.

SNMP can use a number of different protocols as its underlying transport mechanism, but the most common transport protocol, and the only one supported by the switch, is UDP. Therefore the IP module must be enabled and properly configured in order to use SNMP. SNMP trap messages are sent to UDP port 162; all other SNMP messages are sent to UDP port 161. The switch's SNMP agent accepts SNMP messages up to the maximum UDP length the switch can receive.

Other transport mappings have been defined (e.g. OSI [RFC 1418], AppleTalk [RFC 1419] and IPX [RFC 1420]), but the standard transport mapping for the Internet (and the one the switch uses) is UDP. The IP module must be enabled and configured correctly. See Chapter 25, IP Addressing and Protocol Commands for detailed descriptions of the commands required to enable and configure IP.

SNMP Versions

The switch supports SNMP version 1 (SNMPv1), SNMP version 2c (SNMPv2c) and SNMP Version 3 (SNMPv3). The three versions operate similarly.

SNMPv2c updated the original protocol, and offered the following main enhancements:

- a new format for trap messages.
- the get-bulk-request PDU allows for the retrieval of large amounts of data, including tables, with one message.
- more error codes mean that error responses to set messages have more detail than is possible with SNMPv1.
- three new exceptions to errors can be returned for get, get-next and get-bulk-request messages. These are: noSuchObject, noSuchInstance, and endOfMibView.

SNMPv3 provides significant enhancements to address the security weaknesses existing in the earlier versions. This is achieved by implementing two new major features:

- Authentication - by using password hashing and time stamping.
- Privacy - by using message encryption.

Support for multiple versions of SNMP is achieved by responding to each SNMP request with a response of the same version. For example, if an SNMPv1 request is sent to the switch, an SNMPv1 response is returned. If an SNMPv2c request is sent, an SNMPv2c response is returned. Therefore, authentication and encryption functions are not invoked when messages are detected as having either an SNMPv1 or SNMPv2c protocol format.

SNMP Messages

The SNMP protocol is termed simple because it has only six operations, or messages—get, get-next, get-response, set, and trap, and SNMPv2c also has the get-bulk-request message. The replies from the managed device are processed by the NMS and generally used to provide a graphical representation of the state of the network. The two major SNMP operations available to a management station for interacting with a client are the get and set operations. The SNMP set operator can lead to security breaches, since SNMP is not inherently very secure. When forced to operate in either SNMPv1 or v2 mode, when operating with older management stations for example, care must be taken in the choice and safe-guarding of community names, which are effectively passwords for SNMP.

Polling versus Event Notification

SNMP employs a polling paradigm. A Network Management Station (NMS) polls the managed device for information as and when it is required, by sending get-request, get-next-request, and/or get-bulk-request PDUs to the managed device. The managed device responds by returning the requested information in a get-response PDU. The NMS may manipulate objects in the managed device by sending a set-request PDU to the managed device.

The only time that a managed device initiates an exchange of information is in the special case of a trap PDU. A managed device may generate a limited set of traps to notify the NMS of critical events that may affect the ability of the NMS to communicate with the managed device or other managed devices on the network, and therefore to “manage” the network. Such events include the restarting or re-initialization of a device, a change in the status of a network link (up or down), or an authentication failure.

Message Format for SNMPv1 and SNMPv2c

Table 62-1: Fields in an SNMP message

Field	Function
Version	The version of the SNMP protocol. The value is version-1 (0) for the SNMP protocol as defined in RFC 1157, or version-2c (1) for the SNMP protocol as defined in RFC 1902.
Community	The name of an SNMP community, for authentication purposes
SNMP PDU	An SNMP Protocol Data Unit (PDU).

Table 62-2: SNMP PDUs

PDU	Function
get-request	Sent by an NMS to an agent, to retrieve the value of an object.
get-next-request	Sent by an NMS to an agent, to retrieve the value of the next object in the sub-tree. A sub-tree is traversed by issuing a get-request PDU followed by successive get-next-request PDUs.
get-bulk-request	Sent by an NMS to an agent to request a large amount of data with a single message. This is for SNMPv2c messages.
set-request	Sent by an NMS to an agent, to manipulate the value of an object. SNMP PDU Version Community
get-response	Sent by an agent to an NMS in response to a get-request, get-next-request, get-bulk-response, or set-request PDU.
trap	Sent by an agent to an NMS to notify the NMS of a extraordinary event.
report	Although not explicitly defined in the RFCs, reports are used for specific purposes such as EngineID discovery and time synchronization.

Table 62-3: Generic SNMP traps

Value	Meaning
coldStart	The agent is re-initializing itself. Objects may be altered.
warmStart	The agent is re-initializing itself. Objects are not altered.
linkDown	An interface has changed state from up to down.
linkUp	An interface has changed state from down to up.
authenticationFailure	An SNMP message has been received with an invalid community name.
egpNeighborLoss	An EGP peer has transitioned to down state.

SNMP Communities (Version v1 and v2c)

A community is a relationship between an NMS and an agent. The community name is used like a password for a trivial authentication scheme. Both SNMPv1 and SNMPv2c provide security based on the community name only. The concept of communities does not exist for SNMPv3, which instead provides for a far more secure communications method using entities, users, and groups.

Caution **We strongly recommend removing community membership from all SNMPv3 configured devices to prevent access to them via SNMPv1 and SNMv2c, which could bypass the additional SNMPv3 security features.**

SNMPv3 Entities

Entities comprise one of the basic components of the SNMPv3 enhanced architecture. They define the functionality and internal structure of the SNMP managers and agents. An in-depth description of entities can be found in RFC 3411, on which the following text is based. SNMPv3 defines two entity types, a manager and an agent. Both entity types contain two basic components: an SNMP engine and a set of applications.

SNMP Engine

The engine provides the basic services to support the agents component applications, in this respect it performs much of the functionality expected of the ISO Session and Presentation layers. These functions include message transmission and reception, authentication and encryption, and access control to its managed objects database (MIB). The SNMP engine comprises the following components:

- Dispatcher
- Message processing Subsystem
- Security Subsystem
- Access Control Subsystem

The only security subsystem presently supported is the user based security model (USM).

Each SNMP engine is identified by an snmpEngineID that must be unique within the management system. A one to one association exists between an engine and the entity that contains it.

Entity Applications

The following applications are defined within the agent applications:

- Command Generator
- Notification Receiver
- Proxy Forwarder
- Command Responder
- Notification Originator
- Other

SNMPv3 Message Protocol Format

Table 62-4: SNMPv3 PDUs

Value	Meaning
msgVersion	Identifies the message format to be SNMPv3.
msgID	An identifier used between SNMP entities to coordinate message requests and responses. Note that a message response takes the msgID value of the initiating message.
msgMaxSize	Conveys the maximum message size (in octets) an integer between 484 and 2^{31}-1, supported by the sender of the message. Specified as msgFlags. A single octet whose last three bits indicate the operational mode for privacy, authentication, and report.
msgSecurityModel	An identifier used to indicate the security mode (i.e. SNMPv1, SNMPv2c or SNMPv3 to be used when processing the message. Note that although only the SNMPv3 identifier is accepted by the switch, these earlier version message formats are detected by the msgVersion field and processed appropriately.
msgAuthoritativeEngineID	The ID of the authoritative engine that relates to a particular message, i.e. the source engine ID for Traps, Responses and Reports, and the destination engine for Gets, GetNexts, Sets, and Informs.
msgAuthoritativeEngineBoots	A value that represents the number of times the authoritative engine has rebooted since its installation. Its value has the range 1 to 2^{31}-1.
msgAuthoritativeEngineTime	The number of seconds since the authoritative engine snmpEngineBoots counter was last incremented.
msgUserName	The name of the user (principal) on whose behalf the message is being exchanged.
msgAuthenticationParameters	If the message has been authenticated, this field contains a serialized OCTET STRING representing the first 12 octets of the HMAC-MD5-96 output done over the whole message.
msgPrivacyParameters	For encrypted data, this field contains the "salt" used to create the DES encryption Initialization Vector (IV).
ContextEngineID	Within a particular administrative domain, this field uniquely identifies an SNMP entity that may realize an instance of a context with a particular contextName
ContextName	A unique name given to a context within a particular SNMP entity.

SNMPv1 and SNMPv2c

Although software levels 2.6.3 and higher support the specific facilities of SNMP v1 and v2, their documentation is available to provide backward compatibility with older network management systems. The far superior security features offered by implementing SNMPv3 should be used wherever possible.

The switch's implementation of SNMPv1 is based on RFC 1157, *A Simple Network Management Protocol (SNMP)*, and RFC 1812, *Requirements for IP Version 4 Routers*.

When the SNMP agent is disabled, the agent does not respond to SNMP request messages. The agent is disabled by default. The current state and configuration of the SNMP agent can be displayed.

SNMP MIB Views for SNMPv1 and SNMPv2c

An SNMP MIB view is a arbitrary subset of objects in the MIB. Objects in the view may be from any part of the object name space, and not necessarily the same sub-tree. An SNMP community profile is the pairing of an SNMP access mode (read-only or read-write) with the access mode defined by the MIB for each object in the view. For each object in the view, the community profile defines the operations that can be performed on the object.

Pairing an SNMP community with an SNMP community profile determines the level of access that the agent affords to an NMS that is a member of the specified community. When an agent receives an SNMP message, it checks the community name encoded in the message. If the agent knows the community name, the message is deemed to be authentic and the sending SNMP entity is accepted as a member of the community. The community profile associated with the community name then determines the sender's view of the MIB and the operations that can be performed on objects in the view.

SNMP Communities

SNMP communities were introduced into SNMPv1 and retained in version 2c. Although the switch's software still supports communities, this is to provide backward compatibility with legacy management systems. Communities should not be used where a secure network is required. Instead, use the secure network features offered by SNMPv3.

An SNMP community is a pairing of an SNMP agent with a set of SNMP application entities. Communities are the main configuration item in the switch's implementation of SNMPv1 and v2, and are defined in terms of a list of IP addresses which define the SNMP application entities (trap hosts and management stations) in the community.

Important community names act as passwords and provide minimal authentication. Any SNMP application entity that knows a community name can read the value of any instance of any object in the MIB implemented in the switch. Any SNMP application entity that knows the name of a community with write access can change the value of any instance of any object in the MIB implemented in the switch, possibly affecting the operation of the switch. For this reason, take care with the security of community names.

When a trap is generated by the SNMP agent it is forwarded to all trap hosts in all communities. The community name and manager addresses are used to provide trivial authentication. An incoming SNMP message is deemed authentic if it contains a valid community name and originated from an IP address defined as a management station for that community.

When a community is disabled, the SNMP agent behaves as if the community does not exist and generates authentication failure traps for messages directed to the disabled community.

The SNMP agent does not support a default community called "public" with read-only access, traps disabled and open access as mandated in RFC 1812, as this is a security hole open for users who wish to use the switch with minimal modification to the default configuration. The default configuration of the switch has no defined communities. Communities must be explicitly created.

SNMP authentication (for SNMPv1 and v2) is a mechanism whereby an SNMP message is declared to be authentic, that is from an SNMP application entity actually in the community to which the message purports to belong. The mechanism may be trivial or secure. The only form of SNMP authentication implemented by the switch's SNMP agent is trivial authentication. The authentication failure trap may be generated as a result of the failure to authentication an SNMP message.

Switch interfaces can be enabled or disabled via SNMP by setting the ifAdminStatus object in the ifTable of MIB-II MIB to 'Up(1)' or 'Down(2)' for the corresponding ifIndex. If it is not possible to change the status of a particular interface the switch returns an SNMP error message.

The switch's implementation of the ifOperStatus object in the ifTable of MIB-II MIB supports two additional values—"Unknown(4)" and "Dormant(5)" (e.g. an inactive dial-on-demand interface).

Caution **An unauthorized person with knowledge of the appropriate SNMP community name could bring an interface up or down. Community names act as passwords for the SNMP protocol. When creating an SNMP community with write access, take care to select a secure community name and to ensure that only authorized personnel know it.**

An SNMP MIB view is a subset of objects in the MIB that pertain to a particular network element. For example, the MIB view of a hub would be the objects relevant to management of the hub, and would not include IP routing table objects, for example. The switch's SNMP agent does not allow the construction of MIB views. The switch supports all relevant objects from all MIBs that it implements.

Note that the switch's standard set and show commands can also be used to access objects in the MIBs supported by the switch.

Defining Management Stations within Communities

You can add management stations to a community either individually, by entering just its IP address, or you can enter a range of management stations by entering an IP address that ends with a '/' character followed by a number between 1 and 32. The number that follows the '/' character operates as an address mask to define a range of addresses for the management stations. The following example shows how to allocate a band of three binary addresses to a portion of the subnet 146.15.1.X

Example

In this example we make provision for up to 8 possible management stations within a community called "admin".

Step 1:

Decide on the number of management stations that you want to assign to a particular subnet, then decide how many binary digits are required to define this number of addresses. In this case we need up to 8 management stations, so we will assign 3 binary digits (3 binary digits can provide 8 different values). To assign the last 3 binary digits for management stations, we assign a prefix that is a count of all binary digits in the address minus those to be assigned as management stations. In this case the prefix is 29; this being the number of binary digits in an IP address (32) minus the number of digits assigned to the management stations (3).

Step 2:

The method used in this step depends on whether or not the community already exists.

- If the community called "admin" does not exist, create a new community called "admin" and allocate a three binary digit block of addresses to the address subnet 146.15.1.X.
- If the community called "admin" already exists, allocate a three binary digit block of addresses to an existing community called "admin" with the address subnet 146.15.1.X.

For security reasons, the common management prefix should be larger than the IP subnet. This prevents stations on one subnet from being considered valid management stations on a different subnet.

Configuration Example (SNMPv1 and v2)

This example shows how to configure the switch's SNMP agent. Two network management stations have been set up on a large network. The central NMS (IP address 192.168.11.5) monitors devices on the network and uses SNMP set messages to manage devices on the network. Trap messages are sent to this management station. The regional network management station (IP addresses 192.168.16.1) is used just to monitor devices on the network by using SNMP get messages. Link traps are enabled for all interfaces on this particular switch.

IP and VLANs must be correctly configured in order to access the SNMP agent in the switch. This is because the IP module handles both the TCP transport functions, and the UDP functions that enable datagrams to transport SNMP messages. See Chapter 25, IP Addressing and Protocol Commands for commands that enable and configure IP.

To configure SNMP

Step 1: Enable the SNMP agent.

Enable the SNMP agent and enable the generation of authenticate failure traps to monitor unauthorized SNMP access. SNMP is enabled by default in AlliedWare Plus.

```
awplus(config)# snmp-server enable trap auth
```

Step 2: Create a community with write access for the central NMS.

Create a write access community called "example1rw" for use by the central network management station at 192.168.11.5 Use an ACL to give the central NMS SNMP access to the switch using that community name.

```
awplus(config)# access-list 66 permit 192.168.11.5

awplus(config)# snmp-server community example1rw rw 66
```

Care must be taken with the security of community names. Do not use the names "private" or "public" in your network because they are too obvious. Community names act as passwords and provide only trivial authentication. Any SNMP application entity that knows a community name can read the value of any instance of any object in the MIB implemented in the switch. Any SNMP application entity that knows the name of a community with write access can change the value of any instance of any object in the MIB implemented in the switch, possibly affecting the operation of the switch.

SNMP V1 or V2c provide very minimal security. If security is a concern, you should use SNMPv3.

Step 3: Create a community with read-only access for the regional NMS.

Create a read-only access community called "example2ro" for use by the regional network management station at 192.168.16.1. Use an ACL to give the regional NMS SNMP access to the switch using that community name.

```
awplus(config)# access-list 67 permit 192.168.16.1

awplus(config)# snmp-server community example2ro ro 67
```

Step 4: Enable link traps.

Enable link traps for the desired interfaces. In this example, the NSMs are in VLAN 2 and VLAN 3 and other ports are in VLAN 1 for simplicity.

```
   awplus(config)# interface vlan1-3

awplus(config-if)# snmp trap link-status
```

Note that link traps on VLANs are sent when the last port in the VLAN goes down. You will only see a trap for a VLAN if the trap host is in a different VLAN.

You can also enable link traps on channel groups and switch ports. For example, to enable traps on a range of switch ports:

```
   awplus(config)# int port1.0.5-1.0.7

awplus(config-if)# snmp trap link-status
```

You can also enable link traps on channel groups and switch ports. For example, to enable traps on a range of switch ports:

Step 5: Configure trap hosts.

Specify the IP address or addresses that the traps will get sent to. In this example, traps will be sent to both NMSes.

```
awplus(config)# snmp-server host 192.168.11.5 version 2c
                example1rw

awplus(config)# snmp-server host 192.168.16.1 version 2c
                example2ro
```

Step 6: Check the configuration.

Check that the current configuration of the SNMP communities matches the desired configuration:

```
awplus# show snmp-server

awplus# show snmp-server community

awplus# show run snmp
```

This is the output of the **show snmp-server community** command for this example:

```
SNMP community information:
  Community Name ........... example1rw
    Access ................. Read-write
    View ................... none
  Community Name ........... example2ro
    Access ................. Read-only
    View ................... none
```

This is the output of the **show run snmp** command for this example:

```
no snmp-server ipv6
snmp-server enable trap auth
snmp-server community example1rw rw 66
snmp-server community example2ro 67
snmp-server host 192.168.1.2 version 2c example1rw
snmp-server host 192.168.2.2 version 2c example2ro
!
```

Check that the interface link up/down traps have been correctly configured:

```
awplus# show interface vlan1-3
```

This is the output of the **show interface** command for this example:

```
Interface vlan1
  Scope: both
  Link is UP, administrative state is UP
  Hardware is VLAN, address is 0009.41fd.c029
  index 201 metric 1 mtu 1500
  arp ageing timeout 300
  <UP,BROADCAST,RUNNING,MULTICAST>
  VRF Binding: Not bound
  SNMP link-status traps: Sending (suppressed after 20 traps in 60 sec)
  Bandwidth 1g
    input packets 4061, bytes 277043, dropped 0, multicast   packets 3690
    output packets 190, bytes 18123, multicast packets 0 broadcast packets 0
Interface vlan2
  Scope: both
  Link is DOWN, administrative state is UP
  Hardware is VLAN, address is 0009.41fd.c029
  IPv4 address 192.168.11.50/24 broadcast 192.168.11.255
  index 202 metric 1 mtu 1500
  arp ageing timeout 300
  <UP,BROADCAST,MULTICAST>
  VRF Binding: Not bound
  SNMP link-status traps: Sending (suppressed after 20 traps in 60 sec)
  Bandwidth 1g
    input packets 568, bytes 42309, dropped 0, multicast packets 0
    output packets 183, bytes 18078, multicast packets 0 broadcast packets 0
Interface vlan3
  Scope: both
  Link is DOWN, administrative state is UP
  Hardware is VLAN, address is 0009.41fd.c029
  IPv4 address 192.168.16.50/24 broadcast 192.168.16.255
  index 203 metric 1 mtu 1500
  arp ageing timeout 300
  <UP,BROADCAST,MULTICAST>
  VRF Binding: Not bound
  SNMP link-status traps: Sending (suppressed after 20 traps in 60 sec)
    input packets 0, bytes 0, dropped 0, multicast packets 0
    output packets 0, bytes 0, multicast packets 0 broadcast packets 0
```

SNMPv3

SNMPv3 is the third version of the Simple Network Management Protocol. The architecture comprises the following:

- entities that may be either managers, agents, or both
- a management information base (MIB)
- a transport protocol

At least one manager node runs the SNMP management software in every configuration. Managed devices such as routers, servers, and workstations are equipped with an agent software module. The agent provides access to local objects in the MIB that reflect activity and resources at the node. The agent also responds to manager commands to retrieve values from, and set values in the MIB.

SNMP MIB Views for SNMPv3

An SNMP MIB view is a arbitrary subset of objects in the MIB. Objects in the view may be from any part of the object name space, and not necessarily the same sub-tree.

SNMP Groups

Groups were introduced as part of SNMPv3. They are the means by which users are assigned their views and access control policy. Once a group has been created, users can be added to them. In practice a number of groups would be created, each with varying views and access security requirements. Users would then be added to their most appropriate groups. Each Group name and Security Level pair must be unique within a switch.

SNMP Users

Users were introduced as part of SNMPv3. From a system perspective a user is represented as an entity stored in a table that defines the access and authentication criteria to be applied to access or modify the SNMP MIB data.

SNMP Target Addresses

Target addresses were introduced as part of SNMPv3. They specify the destination and user that receives outgoing notifications such as trap messages. SNMP target address names must be unique within the managed device.

SNMP Target Params

Target params were introduced as part of SNMPv3. They specify an entry in the snmpTargetParamsTable. SNMP target params names must be unique within the managed device.

Configuration Example (SNMPv3)

This example shows how to configure the switch's SNMP agent. Two network management stations have been set up on a large network. The central NMS (IP address 192.168.11.5) monitors devices on the network and uses SNMP set messages to manage devices on the network. Trap messages are sent to this management station.

The IP module must be enabled and correctly configured in order to access the SNMP agent in the switch, since the IP module handles the UDP datagrams used to transport SNMP messages.

To configure SNMP

Step 1: Enable the SNMP agent.

Enable the SNMP agent and enable the generation of authenticate failure traps to monitor unauthorized SNMP access. SNMP is enabled by default in AlliedWare Plus.

Step 2: Add SNMP views.

You can specify views using their OID or the predefined MIB name.

```
awplus(config)# snmp-server view atmib 1.3.6.1.2.14
                included

awplus(config)# snmp-server view atmib alliedtelesis
                included
```

Step 3: Add SNMP group.

```
awplus(config)# snmp-server group ord-user noauth read
                atmib

awplus(config)# snmp-server group admin-user auth read
                atmib write atmib notify atmi
```

Step 4: Add SNMP users.

Add users to the groups by using commands such as:

```
awplus(config)# snmp-server user ken admin-user auth md5
                mercury
```

Step 5: Add SNMP target parameters.

Step 6: Add SNMP target address.

Copy a File to or from a TFTP Server

Use this procedure to copy a file (for example, a software version file) to the switch from a TFTP server, or to copy a file (for example, a configuration file) from the switch to a TFTP server. The MIB objects in this procedure reside in the module atFilev2 { modules 600 }, with object ID 1.3.6.1.4.1.207.8.4.4.4.600**.** For detailed descriptions of the MIB objects used in this procedure, and other file management MIB objects, see "AT-FILEv2-MIB" on page 64.62. Other MIB objects can be used in a similar way for moving and deleting files on the switch.

Table 62-5: Procedure for copying a file to or from a device using a TFTP server

	Do this ...	By setting or reading this MIB object ...	Whose object ID is ...	To this value...
1.	If the source device is part of a stack, set the stack ID. For a standalone switch, keep the default value, 1.	atFilev2SourceStackId	{ atFilev2Operation 1 }	<*stack-id*>
2.	If the destination device is part of a stack, set the stack ID.	atFilev2DestinationStackId	{ atFilev2Operation 4 }	<*stack-id*>
3.	Set the source device.	atFilev2SourceDevice	{ atFilev2Operation 2 }	1 (TFTP) or 2 (Flash)
4.	Set the destination device.	atFilev2DestinationDevice	{ atFilev2Operation 5 }	1 (TFTP) or 2 (Flash)
5.	Set the source filename. Include the path (if any) but not the device.	atFilev2SourceFileName	{ atFilev2Operation 3 }	<*source-filename*> e.g. /awp/config/admin.cfg
6.	Set the destination filename. Include the path (if any) but not the device.	atFilev2DestinationFileName	{ atFilev2Operation 6 }	<*dest-filename*> e.g. /config/admin.cfg
7.	Set the IP address of the TFTP server.	atFilev2TftpIPAddr	{ atFilev2Tftp_4 1 }	<*ip-addr*>
8.	Check that no other transfer is in progress, and that the required parameters have been set.	atFilev2CopyBegin	{ atFilev2Operation 7 }	Read: idle
9.	Start the file transfer.	atFilev2CopyBegin	{ atFilev2Operation 7 }	Set: 1
10	Monitor file transfer progress.	atFilev2CopyBegin	{ atFilev2Operation 7 }	Read: In progress: copying <*src*> --> <*dst*> *or* Success: copy <*src*> --> <*dst*> success *or* Failure: copy <*src*> --> <*dst*> failure: <*err-msg*>

Upgrade Software and Configuration Files

Use this procedure to upgrade to a new software version and boot configuration file. For detailed descriptions of the MIB objects used in this procedure, and other MIB objects for managing software installation and configuration files, see "AT-SETUP-MIB" on page 64.43.

Table 62-6: Procedure for upgrading to a new software version and boot configuration

	Do this ...	By reading or setting this MIB object ...	Whose object ID is ...	To this value...
1.	Check that you have enough flash memory for the currently running software file, the new software version file, and any configuration scripts required.			
2.	Check the version and name of the software currently running.	currSoftVersion currSoftName	1.3.6.1.4.1.207.8.4.4.4.500.2.1.1 1.3.6.1.4.1.207.8.4.4.4.500.2.1.2	Read: *<software-name>* *<software-version>*
3.	If you do not already have the currently running software as a software version file in flash, save the currently running software with a file name to the flash root.	currSoftSaveAs	1.3.6.1.4.1.207.8.4.4.4.500.2.1.1	Set: *<backup-filename.rel>*
4.	Check that the file saved successfully. (The most common failures result from lack of flash memory space.)	currSoftSaveAs	1.3.6.1.4.1.207.8.4.4.4.500.2.1.3	Read: saving *<filename>* *or* *<filename>* success *or* *<filename>* failure: *<error-message>*
5.	Copy the new software version file to flash memory on the device	See Table 62-5.		
6.	Set the new release file to be the current release that the device will install and run the next time it restarts. Include the path.	nextBootPath	1.3.6.1.4.1.207.8.4.4.4.500.2.2.2	Set: *<next-filename>* e.g.: `flash:/release.rel`
7.	Check the version of release file set to install next.	nextBootVersion	1.3.6.1.4.1.207.8.4.4.4.500.2.2.1	Read: *<software-version>*
8.	Set the previous release file to be the backup release that the device will install and run if the device fails to boot successfully with the new release file. Include the path.	bckpPath	1.3.6.1.4.1.207.8.4.4.4.500.2.3.2	Set: *<backup-filename>* e.g.: `flash:/release.rel`

Table 62-6: Procedure for upgrading to a new software version and boot configuration(cont.)

	Do this ...	By reading or setting this MIB object ...	Whose object ID is ...	To this value...
9.	Check the version of backup release file.	bckpVersion	1.3.6.1.4.1.207.8.4.4.4.500.2.3.1	Read: *<software-version>*
10.	If necessary, copy a configuration file to the device (Table 62-5), or save the current running configuration to a file in the root directory of flash. To save the running configuration, specify the filename, but not a device or path.	See Table 62-5. or runCnfgSaveAs	1.3.6.1.4.1.207.8.4.4.4.500.3.1.1	Set: *<filename.cfg>* e.g.: myconfig.cfg
11.	Check and if necessary set the file the device will use for configuration when it restarts. Include the full path.	bootCnfgPath	1.3.6.1.4.1.207.8.4.4.4.500.3.2.1	Read/set: *<filename.cfg>* e.g.: flash:/myconfig.cfg
12.	Check that a boot configuration file matching the boot configuration path exists.	bootCnfgExists	1.3.6.1.4.1.207.8.4.4.4.500.3.2.2	Read: TRUE (1) or FALSE (2)
13.	Check that the default configuration file flash:/default.cfg exists.	dfltCnfgExists	1.3.6.1.4.1.207.8.4.4.4.500.3.3.2	Read: TRUE (1) or FALSE (2)
14.	Restart the device.	restartDevice	1.3.6.1.4.1.207.8.4.4.4.500.1	1

Chapter 63: SNMP Commands

Command List

This chapter provides an alphabetical reference for commands used to configure SNMP. For more information, see Chapter 62, SNMP Introduction, and Chapter 64, SNMP MIBs.

For information about modifying or redirecting the output from **show** commands to a file, see "Controlling "show" Command Output" on page 1.41.

debug snmp

This command enables SNMP debugging.

The **no** variant of this command disables SNMP debugging.

Syntax

```
debug snmp [all|detail|error-string|process|receive|send|xdump]
```

```
no debug snmp [all|detail|error-string|process|receive|send|xdump]
```

Parameter	Description
`all`	Enable or disable the display of all SNMP debugging information.
`detail`	Enable or disable the display of detailed SNMP debugging information.
`error-string`	Enable or disable the display of debugging information for SNMP error strings.
`process`	Enable or disable the display of debugging information for processed SNMP packets.
`receive`	Enable or disable the display of debugging information for received SNMP packets.
`send`	Enable or disable the display of debugging information for sent SNMP packets.
`xdump`	Enable or disable the display of hexadecimal dump debugging information for SNMP packets.

Mode Privileged Exec and Global Configuration

Example To start SNMP debugging, use the command:

```
awplus# debug snmp
```

To start SNMP debugging, showing detailed SNMP debugging information, use the command:

```
awplus# debug snmp detail
```

To start SNMP debugging, showing all SNMP debugging information, use the command:

```
awplus# debug snmp all
```

Related Commands show debugging snmp
terminal monitor
undebug snmp

show counter snmp-server

This command displays counters for SNMP messages received by the SNMP agent.

Syntax `show counter snmp-server`

Mode User Exec and Privileged Exec

Example To display the counters for the SNMP agent, use the command:

```
awplus# show counter snmp-server
```

Output Figure 63-1: Example output from the **show counter snmp-server** command

```
SNMP-SERVER counters
inPkts                 ......... 11
inBadVersions          ......... 0
inBadCommunityNames    ......... 0
inBadCommunityUses     ......... 0
inASNParseErrs         ......... 0
inTooBigs              ......... 0
inNoSuchNames          ......... 0
inBadValues            ......... 0
inReadOnlys            ......... 0
inGenErrs              ......... 0
inTotalReqVars         ......... 9
inTotalSetVars         ......... 0
inGetRequests          ......... 2
inGetNexts             ......... 9
inSetRequests          ......... 0
inGetResponses         ......... 0
inTraps                ......... 0
outPkts                ......... 11
outTooBigs             ......... 0
outNoSuchNames         ......... 2
outBadValues           ......... 0
outGenErrs             ......... 0
outGetRequests         ......... 0
outGetNexts            ......... 0
outSetRequests         ......... 0
outGetResponses        ......... 11
outTraps               ......... 0
UnSupportedSecLevels   ......... 0
NotInTimeWindows       ......... 0
UnknownUserNames       ......... 0
UnknownEngineIDs       ......... 0
WrongDigest            ......... 0
DecryptionErrors       ......... 0
UnknownSecModels       ......... 0
InvalidMsgs            ......... 0
UnknownPDUHandlers     ......... 0
```

Table 63-1: Parameters in the output of the **show counter snmp-server** command

Parameter	Meaning
`inPkts`	The total number of SNMP messages received by the SNMP agent.
`inBadVersions`	The number of messages received by the SNMP agent for an unsupported SNMP version. It drops these messages.The SNMP agent on your device supports versions 1, 2C, and 3.

Table 63-1: Parameters in the output of the **show counter snmp-server** command (cont.)

Parameter	Meaning
`inBadCommunityNames`	The number of messages received by the SNMP agent with an unrecognized SNMP community name. It drops these messages.
`inBadCommunityUses`	The number of messages received by the SNMP agent where the requested SNMP operation is not permitted from SNMP managers using the SNMP community named in the message.
`inASNParseErrs`	The number of ASN.1 or BER errors that the SNMP agent has encountered when decoding received SNMP Messages.
`inTooBigs`	The number of SNMP PDUs received by the SNMP agent where the value of the error-status field is 'tooBig'. This is sent by an SNMP manager to indicate that an exception occurred when processing a request from the agent.
`inNoSuchNames`	The number of SNMP PDUs received by the SNMP agent where the value of the error-status field is 'noSuchName'. This is sent by an SNMP manager to indicate that an exception occurred when processing a request from the agent.
`inBadValues`	The number of SNMP PDUs received by the SNMP agent where the value of the error-status field is 'badValue'. This is sent by an SNMP manager to indicate that an exception occurred when processing a request from the agent.
`inReadOnlys`	The number of valid SNMP PDUs received by the SNMP agent where the value of the error-status field is 'readOnly'. The SNMP manager should not generate a PDU which contains the value 'readOnly' in the error-status field. This indicates that there is an incorrect implementations of the SNMP.
`inGenErrs`	The number of SNMP PDUs received by the SNMP agent where the value of the error-status field is 'genErr'.
`inTotalReqVars`	The number of MIB objects that the SNMP agent has successfully retrieved after receiving valid SNMP Get-Request and Get-Next PDUs.
`inTotalSetVars`	The number of MIB objects that the SNMP agent has successfully altered after receiving valid SNMP Set-Request PDUs.
`inGetRequests`	The number of SNMP Get-Request PDUs that the SNMP agent has accepted and processed.
`inGetNexts`	The number of SNMP Get-Next PDUs that the SNMP agent has accepted and processed.
`inSetRequests`	The number of SNMP Set-Request PDUs that the SNMP agent has accepted and processed.
`inGetResponses`	The number of SNMP Get-Response PDUs that the SNMP agent has accepted and processed.
`inTraps`	The number of SNMP Trap PDUs that the SNMP agent has accepted and processed.
`outPkts`	The number of SNMP Messages that the SNMP agent has sent.

Table 63-1: Parameters in the output of the **show counter snmp-server** command (cont.)

Parameter	Meaning
`outTooBigs`	The number of SNMP PDUs that the SNMP agent has generated with the value 'tooBig' in the error-status field. This is sent to the SNMP manager to indicate that an exception occurred when processing a request from the manager.
`outNoSuchNames`	The number of SNMP PDUs that the SNMP agent has generated with the value `noSuchName' in the error-status field. This is sent to the SNMP manager to indicate that an exception occurred when processing a request from the manager.
`outBadValues`	The number of SNMP PDUs that the SNMP agent has generated with the value 'badValue' in the error-status field. This is sent to the SNMP manager to indicate that an exception occurred when processing a request from the manager.
`outGenErrs`	The number of SNMP PDUs that the SNMP agent has generated with the value 'genErr' in the error-status field. This is sent to the SNMP manager to indicate that an exception occurred when processing a request from the manager.
`outGetRequests`	The number of SNMP Get-Request PDUs that the SNMP agent has generated.
`outGetNexts`	The number of SNMP Get-Next PDUs that the SNMP agent has generated.
`outSetRequests`	The number of SNMP Set-Request PDUs that the SNMP agent has generated.
`outGetResponses`	The number of SNMP Get-Response PDUs that the SNMP agent has generated.
`outTraps`	The number of SNMP Trap PDUs that the SNMP agent has generated.
`UnSupportedSecLevels`	The number of received packets that the SNMP agent has dropped because they requested a securityLevel unknown or not available to the SNMP agent.
`NotInTimeWindows`	The number of received packets that the SNMP agent has dropped because they appeared outside of the authoritative SNMP agent's window.
`UnknownUserNames`	The number of received packets that the SNMP agent has dropped because they referenced an unknown user.
`UnknownEngineIDs`	The number of received packets that the SNMP agent has dropped because they referenced an unknown snmpEngineID.
`WrongDigest`	The number of received packets that the SNMP agent has dropped because they didn't contain the expected digest value.
`DecryptionErrors`	The number of received packets that the SNMP agent has dropped because they could not be decrypted.
`UnknownSecModels`	The number of messages received that contain a security model that is not supported by the server. Valid for SNMPv3 messages only.

Table 63-1: Parameters in the output of the **show counter snmp-server** command (cont.)

Parameter	Meaning
`InvalidMsgs`	The number of messages received where the security model is supported but the authentication fails. Valid for SNMPv3 messages only.
`UnknownPDUHandlers`	The number of times the SNMP handler has failed to process a PDU. This is a system debugging counter.

Related Commands show snmp-server

show debugging snmp

This command displays whether SNMP debugging is enabled or disabled.

Syntax

```
show debugging snmp
```

Mode User Exec and Privileged Exec

Example To display the status of SNMP debugging, use the command:

```
awplus# show debugging snmp
```

Output Figure 63-2: Example output from the **show debugging snmp** command

```
Snmp (SMUX) debugging status:
  Snmp debugging is on
```

Related Commands debug snmp

show running-config snmp

This command displays the current configuration of SNMP on your device.

Syntax

```
show running-config snmp
```

Mode Privileged Exec

Example To display the current configuration of SNMP on your device, use the command:

```
awplus# show running-config snmp
```

Output Figure 63-3: Example output from the **show running-config snmp** command

```
snmp-server contact AlliedTelesis
snmp-server location Philippines
snmp-server group grou1 auth read view1 write view1 notify view1
snmp-server view view1 1 included
snmp-server community public
snmp-server user user1 group1 auth md5 password priv des
password
```

Related Commands show snmp-server

show snmp-server

This command displays the status and current configuration of the SNMP server.

Syntax

```
show snmp-server
```

Mode Privileged Exec

Example To display the status of the SNMP server, use the command:

```
awplus# show snmp-server
```

Output Figure 63-4: Example output from the **show snmp-server** command

```
SNMP Server .......................... Enabled
IP Protocol .......................... IPv4
SNMPv3 Engine ID (configured name) ... Not set
SNMPv3 Engine ID (actual) ............ 0x80001f888021338e4747b8e607
```

Related Commands

debug snmp
show counter snmp-server
snmp-server
snmp-server engineID local
snmp-server engineID local reset

show snmp-server community

This command displays the SNMP server communities configured on the device. SNMP communities are specific to v1 and v2c.

Syntax

```
show snmp-server community
```

Mode

Privileged Exec

Example

To display the SNMP server communities, use the command:

```
awplus# show snmp-server community
```

Output

Figure 63-5: Example output from the **show snmp-server community** command

```
SNMP community information:
  Community Name ........... public
    Access ................. Read-only
    View ................... none
```

Related Commands

show snmp-server
snmp-server community

show snmp-server group

This command displays information about SNMP server groups. This command is used with SNMP version 3 only.

Syntax

```
show snmp-server group
```

Mode Privileged Exec

Example To display the SNMP groups configured on the device, use the command:

```
awplus# show snmp-server group
```

Output Figure 63-6: Example output from the **show snmp-server group** command

```
SNMP group information:
  Group name .............. guireadgroup
    Security Level ........ priv
    Read View ............. guiview
    Write View ............ none
    Notify View ........... none

  Group name .............. guiwritegroup
    Security Level ........ priv
    Read View ............. none
    Write View ............ guiview
    Notify View ........... none
```

Related Commands show snmp-server
snmp-server group

show snmp-server user

This command displays the SNMP server users and is used with SNMP version 3 only.

Syntax

```
show snmp-server user
```

Mode Privileged Exec

Example To display the SNMP server users configured on the device, use the command:

```
awplus# show snmp-server user
```

Output Figure 63-7: Example output from the **show snmp-server user** command

```
Name                Group name          Auth         Privacy
-------             -------------       -------      ----------
freddy              guireadgroup        none         none
```

Related Commands show snmp-server
snmp-server user

show snmp-server view

This command displays the SNMP server views and is used with SNMP version 3 only.

Syntax `show snmp-server view`

Mode Privileged Exec

Example To display the SNMP server views configured on the device, use the command:

```
awplus# show snmp-server view
```

Output Figure 63-8: Example output from the **show snmp-server view** command

```
SNMP view information:
  View Name ............... view1
    OID .................... 1
    Type ................... included
```

Related Commands show snmp-server
snmp-server view

snmp trap link-status

Use this command to enable SNMP to send link status notifications (traps) for the interfaces when an interface goes up (linkUp) or down (linkDown).

Use the **no** variant of this command to disable the sending of link status notifications.

Syntax

```
snmp trap link-status

no snmp trap link-status
```

Default By default, link status notifications are disabled.

Mode Interface Configuration

Usage The link status notifications can be enabled for the following interface types:

- switch port (e.g. port 1.0.1)
- VLAN (e.g. vlan2)
- static and dynamic link aggregation (e.g. sa2, po3)

To specify where notifications are sent, use the snmp-server host command on page 63.28. To configure the switch globally to send other notifications, use the snmp-server enable trap command on page 63.22.

Examples To enable SNMP to send link status notifications for ports 1.0.2 to 1.0.12, use following commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.2-1.0.12
awplus(config-if)# snmp trap link-status
```

To disable the sending of link status notifications for port 1.0.2, use following commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.2
awplus(config-if)# no snmp trap link-status
```

Related Commands

show interface
snmp trap link-status suppress
snmp-server enable trap
snmp-server host

snmp trap link-status suppress

Use this command to enable the suppression of link status notifications (traps) for the interfaces beyond the specified threshold, in the specified interval.

Use the **no** variant of this command to disable the suppression of link status notifications for the ports.

Syntax

```
snmp trap link-status suppress
   {time {<1-60>|default}|threshold {<1-20>|default}}
```

```
no snmp trap link-status suppress
```

Parameter	Description
`time`	Set the suppression timer for link status notifications.
`<1-60>`	The suppress time in seconds.
`default`	The default suppress time in seconds (60).
`threshold`	Set the suppression threshold for link status notifications. This is the number of link status notifications after which to suppress further notifications within the suppression timer interval.
`<1-20>`	The number of link status notifications.
`default`	The default number of link status notifications (20).

Default By default, if link status notifications are enabled (they are enabled by default), the suppression of link status notifications is enabled: notifications that exceed the notification threshold (default 20) within the notification timer interval (default 60 seconds) are not sent.

Mode Interface Configuration

Usage An unstable network can generate many link status notifications. When notification suppression is enabled, a suppression timer is started when the first link status notification of a particular type (linkUp or linkDown) is sent for an interface. If the threshold number of notifications of this type is sent before the timer reaches the suppress time, any further notifications of this type generated for the interface during the interval are not sent. At the end of the interval, the sending of link status notifications resumes, until the threshold is reached in the next interval.

Examples To enable the suppression of link status notifications for ports 1.0.2 to 1.0.12 after 10 notifications have been sent in 40 seconds, use following commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.2-1.0.12
awplus(config-if)# snmp trap link-status suppress time 40
                   threshold 10
```

To disable the suppression link status notifications for port 1.0.2, use following commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.2
awplus(config-if)# no snmp trap link-status suppress
```

Related Commands show interface
snmp trap link-status

snmp-server

Use this command to enable the SNMP agent (server) on the switch. The SNMP agent receives and processes SNMP packets sent to the switch, and generates notifications (traps) that have been enabled by the **snmp-server enable trap** command on page 63.22.

Use the **no** variant of this command to disable the SNMP agent on the switch. When SNMP is disabled, SNMP packets received by the switch are discarded, and no notifications are generated. This does not remove any existing SNMP configuration.

Syntax

```
snmp-server [ip|ipv6]
```

```
no snmp-server [ip|ipv6]
```

Parameter	Description
ip	Enable or disable the SNMP agent for IPv4.
ipv6	Enable or disable the SNMP agent for IPv6.

Default By default, the SNMP agent is enabled for both IPv4 and IPv6. If neither the **ip** parameter nor the **ipv6** parameter is specified for this command, then SNMP is enabled or disabled for both IPv4 and IPv6.

Mode Global Configuration

Examples To enable SNMP on the switch for both IPv4 and IPv6, use the commands:

```
        awplus# configure terminal
awplus(config)# snmp-server
```

To enable the SNMP agent for IPv4 on the device, use the commands:

```
        awplus# configure terminal
awplus(config)# snmp-server ip
```

To disable the SNMP agent for both IPv4 and IPv6 on the switch, use the commands:

```
        awplus# configure terminal
awplus(config)# no snmp-server
```

To disable the SNMP agent for IPv4, use the commands:

```
awplus(config)# no snmp-server ipv4
```

Related Commands

show snmp-server
show snmp-server community
show snmp-server user
snmp-server community
snmp-server contact
snmp-server enable trap
snmp-server engineID local
snmp-server group
snmp-server host
snmp-server location
snmp-server view

snmp-server community

This command creates an SNMP community, optionally setting the access mode for the community. The default access mode is read only. If view is not specified, the community allows access to all the MIB objects. The SNMP communities are only valid for SNMPv1 and v2c and provide very limited security. Communities should not be used when operating SNMPv3.

The **no** variant of this command removes an SNMP community. The specified community must already exist on the device.

Syntax

```
snmp-server community <community-name>
    {view <view-name>|ro|rw|<access-list>}

no snmp-server community <community-name> [{view <view-name>|<access-
    list>}]
```

Parameter	Description
`<community-name>`	Community name. The community name is a string up to 20 characters long and is case sensitive.
`view`	Configure SNMP view. If view is not specified, the community allows access to all the MIB objects.
`<view-name>`	View name. The view name is a string up to 20 characters long and is case sensitive.
`ro`	Read-only community.
`rw`	Read-write community.
`<access-list>`	<1-99> Access list number.

Mode Global Configuration

Example The following command creates an SNMP community called "public" with read only access to all MIB variables from any management station.

```
        awplus# configure terminal
awplus(config)# snmp-server community public ro
```

The following command removes an SNMP community called "public"

```
        awplus# configure terminal
awplus(config)# no snmp-server community public
```

Related Commands

show snmp-server
show snmp-server community
snmp-server view

snmp-server contact

This command sets the contact information for the system. The contact name is:

- displayed in the output of the show system command
- stored in the MIB object sysContact

The **no** variant of this command removes the contact information from the system.

Syntax

```
snmp-server contact <contact-info>

no snmp-server contact
```

Parameter	Description
<contact-info>	The contact information for the system, from 0 to 255 characters long. Valid characters are any printable character and spaces.

Mode Global Configuration

Example To set the system contact information to "support@alliedtelesis.co.nz", use the command:

```
        awplus# configure terminal
awplus(config)# snmp-server contact support@alliedtelesis.co.nz
```

Related Commands

show system
snmp-server location
snmp-server group

snmp-server enable trap

Use this command to enable the switch to send the specified notifications (traps).

Note that the Environmental Monitoring traps are enabled by default. So you do not need to issue this command for the Environmental Monitoring traps since these are enabled by default. SNMP environmental monitoring traps defined in AT-ENVMONv2-MIB are enabled by default.

Use the **no** variant of this command to disable the sending of the specified notifications.

Syntax

```
snmp-server enable trap {[auth] [dhcpsnooping] [epsr] [lldp]
    [loopprot] [mstp] [nsm] [pim] [rmon] [vcs] [vrrp]}
```

```
no snmp-server enable trap {[auth] [dhcpsnooping] [epsr] [lldp]
    [loopprot] [mstp] [nsm] [pim] [rmon] [vcs] [vrrp]}
```

Parameter	Description
`auth`	Authentication failure.
`dhcpsnooping`	DHCP snooping and ARP security traps. These notifications must also be set using the ip dhcp snooping violation command on page 53.24, and/or the arp security violation command on page 53.4.
`epsr`	EPSR traps.
`lldp`	Link Layer Discovery Protocol (LLDP) traps. These notifications must also be enabled using the lldp notifications command on page 66.14, and/or the lldp med-notifications command on page 66.9.
`loopprot`	Loop Protection traps.
`mstp`	MSTP traps.
`nsm`	NSM traps.
`pim`	PIM traps.
`rmon`	RMON traps.
`vcs`	VCS traps.
`vrrp`	Virtual Router Redundancy (VRRP) traps.

Default By default, no notifications are generated.

Mode Global Configuration

Usage This command cannot be used to enable link status notifications globally. To enable link status notifications for particular interfaces, use the snmp trap link-status command.

To specify where notifications are sent, use the snmp-server host command.

Examples To enable the device to send PoE related traps, use the following commands:

```
        awplus# configure terminal
awplus(config)# snmp-server enable trap power-inline
```

To disable PoE traps being sent out by the switch, use the following commands:

```
        awplus# configure terminal
awplus(config)# no snmp-server enable power-inline
```

Related Commands show snmp-server
show ip dhcp snooping
snmp trap link-status
snmp-server host

snmp-server engineID local

Use this command to configure the SNMPv3 engine ID. The SNMPv3 engine ID is used to uniquely identify the SNMPv3 agent on a switch when communicating with SNMP management clients. Once an SNMPv3 engine ID is assigned, this engine ID is permanently associated with the switch until you change it.

Use the **no** variant of this command to set the user defined SNMPv3 engine ID to a system generated pseudo-random value by resetting the SNMPv3 engine. The **no snmp-server engineID local** command has the same effect as the **snmp-server engineID local default** command. Note that the snmp-server engineID local reset command is used to force the system to generate a new engine ID when the current engine ID is also system generated.

Syntax

```
snmp-server engineID local {<engine-id>|default}
```

```
no snmp-server engineID local
```

Parameter	Description
<engine-id>	Specify SNMPv3 Engine ID value, a string of up to 27 characters.
default	Set SNMPv3 engine ID to a system generated value by resetting the SNMPv3 engine, provided the current engine ID is user defined. If the current engine ID is system generated, use the snmp-server engineID local reset command to force the system to generate a new engine ID.

Mode Global Configuration

Usage All switches must have a unique engine ID which is permanently set unless it is configured by the user.

In a stacked environment, if the same engine ID was automatically generated for all members of the stack, conflicts would occur if the stack was dismantled. Therefore, each member of the stack will generate its own engine ID and the stack master's ID is used when transmitting SNMPv3 packets. Should a master failover occur, a different engine ID is transmitted. You can modify this behavior by manually assigning all stack members the same engine ID using the snmp-server engineID local command. However, should you decide to separate the stack and use the switches individually, you must remember to change or remove this configuration to prevent conflicts.

Example To set the SNMPv3 engine ID to 800000cf030000cd123456, use the following commands:

```
        awplus# configure terminal
awplus(config)# snmp-server engineID local
                800000cf030000cd123456
```

To set a user defined SNMPv3 engine ID back to a system generated value, use the following commands:

```
        awplus# configure terminal
awplus(config)# no snmp-server engineID local
```

Output The following example shows the engine ID values after configuration:

```
awplus(config)#snmp-server engineid local asdgdfh231234d
awplus(config)#exit
awplus#show snmp-server

SNMP Server .......................... Enabled
IP Protocol .......................... IPv4
SNMPv3 Engine ID (configured name) ... asdgdfh231234d
SNMPv3 Engine ID (actual) ............ 0x80001f888029af52e149198483

awplus(config)#no snmp-server engineid local
awplus(config)#exit
awplus#show snmp-server

SNMP Server .......................... Enabled
IP Protocol .......................... IPv4
SNMPv3 Engine ID (configured name) ... Not set
SNMPv3 Engine ID (actual) ............ 0x80001f888029af52e149198483
```

Validation Commands show snmp-server

Related Commands snmp-server engineID local reset
snmp-server group

snmp-server engineID local reset

Use this command to force the switch to generate a new pseudo-random SNMPv3 engine ID by resetting the SNMPv3 engine. If the current engine ID is user defined, use the snmp-server engineID local command to set SNMPv3 engine ID to a system generated value.

Syntax

```
snmp-server engineID local reset
```

Mode Global Configuration

Example To force the SNMPv3 engine ID to be reset to a system generated value, use the commands:

```
        awplus# configure terminal
awplus(config)# snmp-server engineID local reset
```

Validation Commands show snmp-server

Related Commands snmp-server engineID local

snmp-server group

This command is used with SNMP version 3 only, and adds an SNMP group, optionally setting the security level and view access modes for the group. The security and access views defined for the group represent the minimum required of its users in order to gain access.

The **no** variant of this command deletes an SNMP group, and is used with SNMPv3 only. The group with the specified authentication/encryption parameters must already exist.

Syntax

```
snmp-server group <groupname> {auth|noauth|priv}
   [read <readname>|write <writename>|notify <notifyname>]

no snmp-server group <groupname> {auth|noauth|priv}
```

Parameter	Description
<groupname>	Group name. The group name is a string up to 20 characters long and is case sensitive.
`auth`	Authentication.
`noauth`	No authentication and no encryption.
`priv`	Authentication and encryption.
`read`	Configure read view.
<readname>	Read view name.
`write`	Configure write view.
<writename>	Write view name. The view name is a string up to 20 characters long and is case sensitive.
`notify`	Configure notify view.
<notifyname>	Notify view name. The view name is a string up to 20 characters long and is case sensitive.

Mode Global Configuration

Examples To add SNMP group, for ordinary users, user the following commands:

```
        awplus# configure terminal
awplus(config)# snmp-server group usergroup noauth read
                useraccess write useraccess
```

To delete SNMP group `usergroup`, use the following commands

```
        awplus# configure terminal
awplus(config)# no snmp-server group usergroup noauth
```

Related Commands

snmp-server
show snmp-server
show snmp-server group
show snmp-server user

snmp-server host

This command specifies an SNMP trap host destination to which Trap or Inform messages generated by the device are sent.

For SNMP version 1 and 2c you must specify the community name parameter. For SNMP version 3, specify the authentication/encryption parameters and the user name. If the version is not specified, the default is SNMP version 1. Inform messages can be sent instead of traps for SNMP version 2c and 3.

Use the **no** variant of this command to remove an SNMP trap host. The trap host must already exist.

The trap host is uniquely identified by:

- host IP address (IPv4 or IPv6),
- inform or trap messages,
- community name (SNMPv1 or SNMP v2c) or the authentication/encryption parameters and user name (SNMP v3).

Syntax

```
snmp-server host {<ipv4-address>|<ipv6-address>} [traps] [version 1]
    <community-name>]

snmp-server host {<ipv4-address>|<ipv6-address>} [informs|traps]
    version 2c <community-name>

snmp-server host {<ipv4-address>|<ipv6-address>} [informs|traps]
    version 3 {auth|noauth|priv} <user-name>

no snmp-server host {<ipv4-address>|<ipv6-address>} [traps]
    [version 1] <community-name>

no snmp-server host {<ipv4-address>|<ipv6-address>} [informs|traps]
    version 2c <community-name>

no snmp-server host {<ipv4-address>|<ipv6-address>} [informs|traps]
    version 3 {auth|noauth|priv} <user-name>
```

Parameter	Description
<ipv4-address>	IPv4 trap host address in the format `A.B.C.D`, for example, `192.0.2.2`.
<ipv6-address>	IPv6 trap host address in the format `x:x::x:x` for example, `2001:db8::8a2e:7334`.
`informs`	Send Inform messages to this host.
`traps`	Send Trap messages to this host (default).
`version`	SNMP version to use for notification messages. Default: version 1.
`1`	Use SNMPv1 (default).
`2c`	Use SNMPv2c.
`3`	Use SNMPv3.
`auth`	Authentication.
`noauth`	No authentication.

Parameter(cont.)	Description(cont.)
`priv`	Encryption.
<*community-name*>	The SNMPv1 or SNMPv2c community name.
<*user-name*>	SNMPv3 user name.

Mode Global Configuration

Examples To configure the device to send generated traps to the IPv4 host destination `192.0.2.5` with the SNMPv2c community name *public*, use the following command:

```
        awplus# configure terminal
awplus(config)# snmp-server host 192.0.2.5 version 2c public
```

To configure the device to send generated traps to the IPv6 host destination `2001:db8::8a2e:7334` with the SNMPv2c community name *private*, use the following command:

```
        awplus# configure terminal
awplus(config)# snmp-server host 2001:db8::8a2e:7334 version 2c
                private
```

To remove a configured trap host of 192.0.2.5 with the SNMPv2c community name *public*, use the following command:

```
        awplus# configure terminal
awplus(config)# no snmp-server host 192.0.2.5 version 2c public
```

Related Commands snmp trap link-status
snmp-server enable trap
snmp-server view

snmp-server location

This command sets the location of the system. The location is:

- displayed in the output of the show system command
- stored in the MIB object sysLocation

The **no** variant of this command removes the configured location from the system.

Syntax

```
snmp-server location <location-name>
```

```
no snmp-server location
```

Parameter	Description
<location-name>	The location of the system, from 0 to 255 characters long. Valid characters are any printable character and spaces.

Mode Global Configuration

Example To set the location to "server room 523", use the following commands:

```
        awplus#  configure terminal
awplus(config)#  system location server room 523
```

Related Commands
show snmp-server
show system
snmp-server contact

snmp-server source-interface

Use this command to specify the interface that SNMP traps or informs originate from. You cannot specify an interface that does not already have an IP address assigned to the interface.

Use the **no** variant of this command to reset to the default source interface that SNMP traps or informs originate from (the Egress interface as sent from by default).

Syntax

```
snmp-server source-interface {traps|informs} <interface-name>

no snmp-server source-interface {traps|informs}
```

Parameter	Description
`traps`	SNMP traps.
`informs`	SNMP informs.
`<interface-name>`	Interface name (with an IP address already assigned).

Default By default the source interface is the Egress interface where traps or informs were sent from.

Mode Global Configuration

Usage An SNMP trap or inform sent from an SNMP server has the notification IP address of the interface where it was sent from. Use this command to monitor notifications from an interface.

Example To set the interface that SNMP informs originate from to port 1.0.2 for inform packets, use the following commands:

```
        awplus# configure terminal
awplus(config)# snmp-server source-interface informs port1.0.2
```

To reset the interface to the default source interface (the Egress interface) that SNMP traps originate from for trap packets, use the following commands:

```
        awplus# configure terminal
awplus(config)# no snmp-server source-interface traps
```

Validation Commands show running-config

snmp-server user

Use this command to create or move users as members of specified groups. This command is used with SNMPv3 only.

The **no** variant of this command removes an SNMPv3 user. The specified user must already exist.

Syntax

```
snmp-server user <username> <groupname> [encrypted]
    [auth {md5|sha} <auth-password>]
    [priv {des|aes} <privacy-password>]
```

```
no snmp-server user <username>
```

Parameter	Description
<username>	User name. The user name is a string up to 20 characters long and is case sensitive.
<groupname>	Group name. The group name is a string up to 20 characters long and is case sensitive.
`encrypted`	Use the encrypted parameter when you want to enter encrypted passwords.
`auth`	Authentication protocol.
`md5`	MD5 Message Digest Algorithms.
`sha`	SHA Secure Hash Algorithm.
<auth-password>	Authentication password. The password is a string of 8 to 20 characters long and is case sensitive.
`priv`	Privacy protocol.
`des`	DES Data Encryption Standard.
`aes`	AES Advanced Encryption Standards.
<privacy-password>	Privacy password. The password is a string of 8 to 20 characters long and is case sensitive.

Mode Global Configuration

Usage Additionally this command provides the option of selecting an authentication protocol and (where appropriate) an associated password. Similarly, options are offered for selecting a privacy protocol and password.

- Note that each SNMP user must be configured on both the manager and agent entities. Where passwords are used, these passwords must be the same for both entities.
- Use the **encrypted** parameter when you want to enter already encrypted passwords in encrypted form as displayed in the running and startup configs stored on the switch. For example, you may need to move a user from one group to another group and keep the same passwords for the user instead of removing the user to apply new passwords.
- User passwords are entered using plaintext without the **encrypted** parameter and are encrypted according to the authentication and privacy protocols selected.
- User passwords are viewed as encrypted passwords in running and startup configs shown

from **show running-config** and **show startup-config** commands respectively. Copy and paste encrypted passwords from running-configs or startup-configs to avoid entry errors.

Examples To add SNMP user `authuser` as a member of group `usergroup`, with authentication protocol `md5`, authentication password `Authpass`, privacy protocol `des` and privacy password `Privpass`, use the following commands

```
        awplus#  configure terminal

 awplus(config)#  snmp-server user authuser usergroup auth md5
                  Authpass priv des Privpass
```

Validate the user is assigned to the group using the following command:

```
awplus#show snmp-server user
Name                Group name          Auth          Privacy
-------             -------------       -------       ----------
authuser            usergroup           md5           des
```

To enter existing SNMP user `authuser` with existing passwords as a member of group `newusergroup` with authentication protocol `md5` plus the encrypted authentication password `0x1c74b9c22118291b0ce0cd883f8dab6b74`, privacy protocol `des` plus the encrypted privacy password `0x0e0133db5453ebd03822b004eeacb6608f`, use the following commands

```
        awplus#  configure terminal

 awplus(config)#  snmp-server user authuser newusergroup
                  encrypted auth md5
                  0x1c74b9c22118291b0ce0cd883f8dab6b74 priv des
                  0x0e0133db5453ebd03822b004eeacb6608f
```

Note Copy and paste the encrypted passwords from the **running-config** or the **startup-config** displayed, using the **show running-config** and **show startup-config** commands respectively, into the command line to avoid key stroke errors issuing this command.

Validate the user has been moved from the first group using the following command:

```
awplus#show snmp-server user
Name                Group name          Auth          Privacy
-------             -------------       -------       ----------
authuser            newusergroup        md5           des
```

To delete SNMP user `authuser`, use the following commands:

```
        awplus#  configure terminal

 awplus(config)#  no snmp-server user authuser
```

Related Commands show snmp-server user
snmp-server view

snmp-server view

Use this command to create an SNMP view that specifies a sub-tree of the MIB. Further sub-trees can then be added by specifying a new OID to an existing view. Views can be used in SNMP communities or groups to control the remote manager's access.

Note The object identifier must be specified in a sequence of integers separated by decimal points.

The **no** variant of this command removes the specified view on the device. The view must already exist.

Syntax

```
snmp-server view <view-name> <mib-name> {included|excluded}
```

```
no snmp-server view <view-name>
```

Parameter	Description
`<view-name>`	SNMP server view name. The view name is a string up to 20 characters long and is case sensitive.
`<mib-name>`	Object identifier of the MIB.
`included`	Include this OID in the view.
`excluded`	Exclude this OID in the view.

Mode Global Configuration

Examples The following command creates a view called "loc" that includes system location mib sub-tree.

```
awplus(config)# snmp-server view loc 1.3.6.1.2.1.1.6.0 included
```

To remove the view "loc" use the following command

```
awplus(config)# no snmp-server view loc
```

Related Commands show snmp-server view
snmp-server community

undebug snmp

This command applies the functionality of the no debug snmp command.

Chapter 64: SNMP MIBs

Introduction

This chapter describes the Management Information Bases (MIBs) and managed objects supported by the AlliedWare Plus™ Operating System. The following topics are covered:

- "Allied Telesis Enterprise MIB" on page 64.5 describes the objects implemented in the Allied Telesis Enterprise MIB
- "Public MIBs" on page 64.74 describes the public MIBs supported by the AlliedWare Plus^{TM} Operating System, and any variations from the standard implementation.

About MIBs

A MIB is a collection of managed objects organized into a tree-like hierarchy of nodes in which the managed objects form the leaves. Within the tree, each node is identified by a non-negative integer identifier that is unique among the node's siblings. The address, or object identifier, of any node within the tree is expressed as a series of dot-delimited node identifiers that trace the path from the root of the tree to the node. For example, the object identifier for the sysDescr object is 1.3.6.1.2.1.1.1.

For more information about MIBs and the structure of management information, see Chapter 62, SNMP Introduction.

About SNMP

A network management station (NMS) uses a protocol known as Simple Network Management Protocol (SNMP) to query or change the values of objects in the MIB of managed devices.

A managed device uses SNMP to respond to queries from an NMS, and to send unsolicited alerts (traps) to an NMS in response to events.

For more information about the Simple Network Management Protocol (SNMP), see Chapter 62, SNMP Introduction.

For information about configuring SNMP, see Chapter 63, SNMP Commands.

Obtaining MIBs

You can download MIBs from the following locations:

Download this MIB...	From this location...
Allied Telesis Enterprise MIB	The MIB files are available with the software files from the Support area at http://www.alliedtelesis.com.
Public MIBs defined in RFCs	http://www.rfc-editor.org/rfc.html
IANAifType-MIB	http://www.iana.org/assignments/ianaiftype-mib

Loading MIBs

Individual MIBs define a portion of the total MIB for a device. For example, the MAU-MIB defines objects for managing IEEE 802.3 medium attachment units (MAUs), and forms a sub-tree under mib-2 with the object identifier snmpDot3MauMgt (1.3.6.1.2.1.26).

All the objects within a MIB are assigned object identifiers relative to a parent object. Most MIBs import the object identifier of the parent object, along with other object identifiers, textual conventions, macros and syntax types from the MIBs where they are defined. This creates dependencies between MIBs.

Some network management stations and MIB compilers will generate errors if you load a MIB that depends on another MIB that has not already been loaded. To avoid these errors, we recommend that you load MIBs in the following order:

1. RFC 1212
   RFC 1239
   RFC 2257
   RFC 3410

2. RFC1155-SMI (RFC 1155)
   SNMPv2-SMI (RFC 2578)
   SNMPv2-PDU (RFC 3416)

3. RFC1213-MIB (RFC 1213)
   RFC 1215
   SNMPv2-TC (RFC 2579)
   SNMPv2-CONF (RFC 2580)

4. IP-MIB (RFC 2011)
   TCP-MIB (RFC 2012)
   UDP-MIB (RFC 2013)
   IP-FORWARD-MIB (RFC 2096)
   SNMP-MPD-MIB (RFC 2572)
   RMON-MIB (RFC 2819)
   HCNUM-TC (RFC 2856)
   SNMP-FRAMEWORK-MIB (RFC 3411)
   SNMP-MPD-MIB (RFC 3412)
   SNMPv2-TM (RFC 3417)
   SNMPv2-MIB (RFC 3418)
   INET-ADDRESS-MIB (RFC 4001)
   IANAifType-MIB

5. IF-MIB (RFC 2863)
   SNMP-TARGET-MIB (RFC 3413)

6. SNMP-COMMUNITY-MIB (RFC 2576)
   EtherLike-MIB (RFC 3635)
   MAU-MIB (RFC 3636)
   BRIDGE-MIB (RFC 4188)
   DISMAN-PING-MIB (RFC 4560)
   SNMP-NOTIFICATION-MIB (RFC 3413)
   SNMP-PROXY-MIB (RFC 3413)

7. P-BRIDGE-MIB (RFC 2674)
   Q-BRIDGE-MIB (RFC 2674)
   RSTP-MIB (RFC 4318)

LLDP-MIB
LLDP-EXT-DOT1-MIB
LLDP-EXT-DOT3-MIB
LLDP-EXT-MED-MIB
POE-MIB
VRRP-MIB

8. AT-SMI-MIB

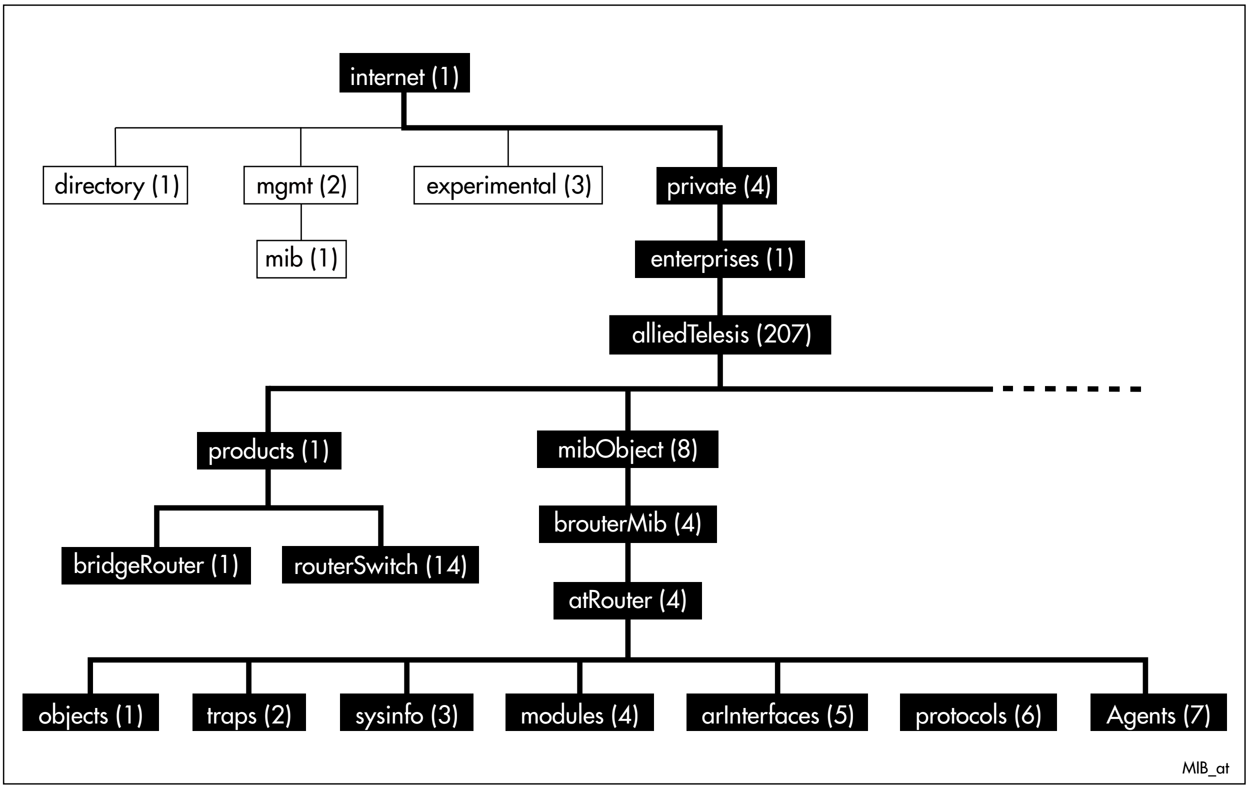
9. AT-BOARDS-MIB
AT-PRODUCT-MIB
AT-SETUP-MIB
AT-SYSINFO-MIB
AT-TRIGGER-MIB
AT-VCSTACK-MIB
AT-USER-MIB
AT-RESOURCE-MIB
AT-LICENSE-MIB
AT-LOOPPROTECT-MIB
AT-DNS-CLIENT--MIB
AT-NTP-MIB
AT-EPSRv2-MIB
AT-FILEv2-MIB
AT-LOG-MIB
AT-IP-MIB
AT-ENVMONv2-MIB
AT-MIBVERSION-MIB
AT-DHCPSN-MIB

Allied Telesis Enterprise MIB

The *Allied Telesis Enterprise MIB* defines a portion of the Management Information Base (MIB) for managing Allied Telesis products and features that are not supported by public MIBs. Objects defined in this MIB reside in the private(4) subtree and have the object identifier alliedTelesis ({ enterprises 207 }).

This document describes only those portions of the Allied Telesis Enterprise MIB supported by the AlliedWare Plus™ Operating System. Figure 64-1 shows the structure of the Allied Telesis Enterprise MIB. Each component MIB is detailed in the following sections of this chapter.

Figure 64-1: The Allied Telesis Enterprise MIB sub-tree of the Internet-standard Management Information Base (MIB)



AT-SMI-MIB

AT-SMI-MIB defines the high-level structure and root objects of the Allied Telesis Enterprise MIB (Table 64-1). These objects are imported by other component MIBs of the Allied Telesis Enterprise MIB.

Table 64-1: AT Enterprise MIB - High Level Structure

Object	Object Identifier	Description
alliedTelesis	{ enterprises 207 } 1.3.6.1.4.1.207	Root of the Allied Telesis Enterprise MIB under the private(4) node defined in RFC1155-SMI.
products	{ alliedTelesis 1 } 1.3.6.1.4.1.207.1	Sub-tree of all product OIDs. Described in "AT-PRODUCTS-MIB" on page 64.9.
bridgeRouter	{ products 1 } 1.3.6.1.4.1.207.1.1	Sub-tree of bridge product MIB objects (not applicable for AlliedWare Plus).
routerSwitch	{ products 14 } 1.3.6.1.4.1.207.1.2	Sub-tree for all router and switch product MIB objects.
mibObject	{ alliedTelesis 8 } 1.3.6.1.4.1.207.8	Sub-tree for all managed objects.
brouterMib	{ mibObject 4 } 1.3.6.1.4.1.207.8.4	Sub-tree of objects for managing bridges, routers, and switches.
atRouter	{ brouterMib 4 } 1.3.6.1.4.1.207.8.4.4	Sub-tree of objects for managing multiprotocol routers and switches.
objects	{ atRouter 1 } 1.3.6.1.4.1.207.8.4.4.1	Sub-tree of OIDs for boards, releases, interface types, and chips.
traps	{ atRouter 2 } 1.3.6.1.4.1.207.8.4.4.2	Sub-tree for generic traps (not applicable for AlliedWare Plus).
sysinfo	{ atRouter 3 } 1.3.6.1.4.1.207.8.4.4.3	Sub-tree of objects describing general system information.
modules	{ atRouter 4 } 1.3.6.1.4.1.207.8.4.4.4	Sub-tree of objects for monitoring and managing software features.
arInterfaces	{ atRouter 5 } 1.3.6.1.4.1.207.8.4.4.5	Sub-tree of objects describing boards, slots and physical interfaces.
protocols	{ atRouter 6 } 1.3.6.1.4.1.207.8.4.4.6	Sub-tree of OIDs for protocols.
atAgents	{ atRouter 7 } 1.3.6.1.4.1.207.8.4.4.7	Sub-tree of objects describing variations from standards.

Table 64-2 lists the major modules of the AT-SMI-MIB grouped by their object identifiers. Note that this is also the order in which they are described in this chapter.

Table 64-2: AT-SMI-MIBs Listed by Object Group

MIB Section	OID	Description
AT-SMI-MIB		This section describes the structure of management information for the Allied Telesis Enterprise object, alliedTelesis { 1.3.6.1.4.1.207 }.
AT-PRODUCTS-MIB	1.3.6.1.4.1.207.1	Object identifiers for Allied Telesis products. See "AT-PRODUCTS-MIB" on page 64.9.
AT-BOARDS-MIB	1.3.6.1.4.1.207.8.4.4.1.1	Object identifiers for boards, interface types, and chip sets. See "AT-BOARDS-MIB" on page 64.11.
AT-SYSINFO-MIB	1.3.6.1.4.1.207.8.4.4.3	Objects that describe generic system information and environmental monitoring. See "AT-SYSINFO-MIB" on page 64.13.
AT-ENVMONv2-MIB	1.3.6.1.4.1.207.8.4.4.3.12	Objects and traps for monitoring fans, voltage rails, temperature sensors, and power supply bays. See "AT-ENVMONv2-MIB" on page 64.18.
AT-VCSTACK-MIB	1.3.6.1.4.1.207.8.4.4.3.13	Objects for managing Virtual Chassis Stacking (VCS). See "AT-VCSTACK-MIB" on page 64.24.
AT-MIBVERSION-MIB	1.3.6.1.4.1.207.8.4.4.3.15	Object to display the last software release that contained changes to the support AT Enterprise MIB definition files. See "AT-MIBVERSION-MIB" on page 64.29.
AT-USER-MIB	1.3.6.1.4.1.207.8.4.4.3.20	Objects for displaying information of users currently logged into a device, or configured in the Local User Data base of the device. See "AT-USER-MIB" on page 64.31.
AT-RESOURCE-MIB	1.3.6.1.4.1.207.8.4.4.3.21	Objects for displaying system hardware resource information. See "AT-RESOURCE-MIB" on page 64.34.
AT-LICENSE-MIB	1.3.6.1.4.1.207.8.4.4.3.22	Objects for managing software licenses on devices using AlliedWare Plus^{TM} Operating System. See "AT-LICENSE-MIB" on page 64.36.
AT-TRIGGER-MIB	1.3.6.1.4.1.207.8.4.4.4.53	Objects for managing triggers. See "AT-TRIGGER-MIB" on page 64.39.
AT-LOOPPROTECT-MIB	1.3.6.1.4.1.207.8.4.4.4.54	Objects for managing Allied Telesis Loop Protection. See "AT-LOOPPROTECT-MIB" on page 64.41.
AT-SETUP-MIB	1.3.6.1.4.1.207.8.4.4.4.500	Objects for managing software installation and configuration files. See "AT-SETUP-MIB" on page 64.43.
AT-DNS-CLIENT-MIB	1.3.6.1.4.1.207.8.4.4.4.501	Objects for managing Allied Telesis DNS Client Configuration. See "AT-DNS-CLIENT-MIB" on page 64.52.
AT-NTP-MIB	1.3.6.1.4.1.207.8.4.4.4.502	Objects for managing Allied Telesis Network Time Protocol (NTP) configuration. See "AT-NTP-MIB" on page 64.53.
AT-EPSRv2-MIB	1.3.6.1.4.1.207.8.4.4.4.536	Objects for managing Allied Telesis EPSR. See "AT-EPSRv2-MIB" on page 64.56.
AT-DHCPSN-MIB	1.3.6.1.4.1.207.8.4.4.4.537	Objects for managing Allied Telesis DHCP Snooping. See "AT-DHCPSN-MIB" on page 64.59.
AT-FILEv2-MIB	1.3.6.1.4.1.207.8.4.4.4.600	Objects for displaying and managing file content on local, stacked and remote sources. See "AT-FILEv2-MIB" on page 64.62.

Table 64-2: AT-SMI-MIBs Listed by Object Group(cont.)

MIB Section	OID	Description
AT-LOG-MIB	1.3.6.1.4.1.207.8.4.4.4.601	Objects for listing log entries from the buffered and permanent logs. See "AT-LOG-MIB" on page 64.68.
AT-IP-MIB	1.3.6.1.4.1.207.8.4.4.4.602	Objects for Allied Telesis specific IP address management. See "AT-IP-MIB" on page 64.70.

AT-PRODUCTS-MIB

Table 64-3: Object identifiers for Allied Telesis products supported by the AlliedWare Plus^{TM} Operating System

Object	Object Identifier	Description
products	{ alliedTelesis 1 }	
routerSwitch	{ products 14 }	Subtree beneath which router and switch product MIB object IDs are assigned.
at_SwitchBladex908	{ routerSwitch 69 }	Switchblade x908 8 Slot Layer 3 Switch Chassis
at_x900_12XTS	{ routerSwitch 70 }	AT-x900-12XT/S Advanced Gigabit Layer 3+ Expandable Switch, 12 x combo ports (10/100/1000BASE-T copper or SFP), 1 x 30Gbps expansion bay
at_x900_24XT	{ routerSwitch 75 }	x900-24XT Enhanced Gigabit Layer 3+ Expandable Switch, 24 x 10/100/1000BASE-T copper ports (RJ-45 connectors), 2 x 20 Gigabit expansion bays
at_x900_24XS	{ routerSwitch 76 }	x900-24XS Enhanced Gigabit Layer 3+ Expandable Switch, 24 x 10/100/1000BASE-T copper ports (RJ-45 connectors), 2 x 20 Gigabit expansion bays
at_x900_24XT_N	{ routerSwitch 77 }	x900-24XT-N Enhanced Gigabit Layer 3+ Expandable Switch, 24 x 10/100/1000BASE-T copper ports (RJ-45 connectors), 2 x 20 Gigabit expansion bays, NEBS compliant
at_x600_24Ts	{ routerSwitch 80 }	x600-24Ts Stackable Managed L2+/L3 Ethernet Switch, 24 x 1000BASE-T copper ports, 4 x SFP (combo) ports
at_x600_24TsXP	{ routerSwitch 81 }	x600-24Ts/XP Stackable Managed L2+/L3 Ethernet Switch, 24 x 1000BASE-T copper ports, 4 x SFP (combo) ports, 2 x XFP ports
at_x600_48Ts	{ routerSwitch 82 }	x600-48Ts Stackable Managed L2+/L3 Ethernet Switch, 48 x 1000BASE-T copper ports, 4 x SFP ports
at_x600_48TsXP	{ routerSwitch 83 }	x600-48Ts/XP Stackable Managed L2+/L3 Ethernet Switch, 48 x 1000BASE-T copper ports, 4 x SFP ports, 2 x XFP ports
at_x600-24TsPoE	{ routerSwitch 91}	x600-24Ts-POE Stackable Managed L2+/L3 Ethernet PoE Switch, 24 x 1000BASE-T PoE ports, 4 x SFP (combo) ports
at_x600_24TPoEPlus	{routerSwitch 92}	x600-24Ts-POE+ Stackable Managed L2+/L3 Ethernet PoE+ Switch, 24 x 1000BASE-T PoE+ ports, 4 x SFP (combo) ports
x610_48Ts_X_POEPlus	{routerSwitch 93}	x610-48Ts/X-POE+ Stackable Managed L2+/L3 Ethernet PoE+ Switch, 48 x 1000BASE-T PoE+ ports, 2 x SFP (combo) ports, 2 x SFP+ ports
x610_48Ts_POEPlus	{routerSwitch 94}	x610-48Ts-POE+ Stackable Managed L2+/L3 Ethernet PoE+ Switch, 48 x 1000BASE-T PoE+ ports, 4 x SFP (combo) ports
x610_24Ts_X_POEPlus	{routerSwitch 95}	x610-24Ts/X-POE+ Stackable Managed L2+/L3 Ethernet PoE+ Switch, 24 x 1000BASE-T PoE+ ports, 4 x SFP (combo) ports, 2 x SFP+ ports
x610_24Ts_POEPlus	{routerSwitch 96}	x610-24Ts-POE+ Stackable Managed L2+/L3 Ethernet PoE+ Switch, 24 x 1000BASE-T PoE+ ports, 4 x SFP (combo) ports
x610_48Ts_X	{routerSwitch 97}	x610-48Ts/X Stackable Managed L2+/L3 Ethernet Switch, 48 x 1000BASE-T copper ports, 2 x SFP (combo) ports, 2 x SFP+ ports

Table 64-3: Object identifiers for Allied Telesis products supported by the AlliedWare Plus^{TM} Operating

Object	Object Identifier	Description
x610_48Ts	{routerSwitch 98}	x610-48Ts Stackable Managed L2+/L3 Ethernet Switch, 24 x 1000BASE-T copper ports, 4 x SFP (combo) ports
x610_24Ts_X	{routerSwitch 99}	x610-24Ts/X Stackable Managed L2+/L3 Ethernet Switch, 24 x 1000BASE-T copper ports, 4 x SFP (combo) ports, 2 x SFP+ ports
x610_24Ts	{routerSwitch 100}	x610-24Ts Stackable Managed L2+/L3 Ethernet Switch, 24 x 1000BASE-T copper ports, 4 x SFP (combo) ports
x610_24SP_X	{routerSwitch 101}	x610-24SP/X Stackable Managed L2+/L3 Ethernet Switch, 24 x SFP (combo) ports, 2 x SFP+ ports
x510_28GTX	{routerSwitch 109}	x510-28GTX Stackable Managed L2+/L3 Ethernet Switch with 24 x 10/100/1000 Base-T ports and 4 x 10 Gb/s SFP+ ports.
x510_28GPX	{routerSwitch 110}	x510-28GPX Stackable Managed L2+/L3 Ethernet Switch with 24 x 10/100/1000 Base-T ports with PoE, 4 x 10 Gb/s SFP+ ports.
x510_28GSX	{routerSwitch 111}	x510-28GSX Stackable Managed L2+/L3 Ethernet Switch with 24 x 100/1000 SFP ports and 4 x 10 Gb/s SFP+ ports.
x510_52GTX	{routerSwitch 112}	x510-52GTX Stackable Managed L2+/L3 Ethernet Switch with 48 x 10/100/1000 Base-T ports and 4 x 10 Gb/s SFP+ ports.
x510_52GPX	{routerSwitch 113}	x510-52GPX Stackable Managed L2+/L3 Ethernet Switch with 48 x 10/100/1000 Base-T ports with PoE, and 4 x 10 Gb/s SFP+ ports.

AT-BOARDS-MIB

- Base CPU and expansion boards (Table 64-4). These object identifiers are for use with the hrDeviceID object in the Host Resources MIB (see "Public MIBs" on page 64.74).
- Interface types (Table 64-5).
- Chip sets (Table 64-6).

Table 64-4: Object identifiers for base CPU and expansion boards

Object	Object Identifier	Description
boards	{ objects 1 }	
pprXem2XP	{ boards 306 }	XEM-2XP Expansion Module, 2 x 10Gbe XFP port
pprXem2XT	{ boards 325 }	XEM-2XT Expansion Module, 2 x 10Gbe copper XEM port
pprPWR800	{ boards 336 }	AT-PWR800, 800W power supply unit
pprPWR1200	{ boards 337 }	AT-PWR1200, 1200W power supply unit
pprPWR250	{ boards 338 }	AT-PWR250, 250W power supply unit
pprPWR250DC	{ boards 351 }	AT-PWR250DC, 250W DC power supply unit
pprx51028GTX	{ boards 370 }	AT-x510-28GTX board with 24 10/100/1000 Base-T ports and four 10Gb/s SFP+ ports.
pprx51028GPX	{ boards 371 }	AT-x510-28GPX board with 24 10/100/1000 Base-T ports, four 10 Gb/s SFP+ ports and PSE function available on pins 1/2 and 3/6 (Mode A) of every copper port.
pprx51028GSX	{ boards 372 }	AT-x510-28GSX with 24 100/1000 SFP ports and four 10 Gb/s SFP+ ports.
pprx51052GTX	{ boards 373 }	AT-x510-52GTX board with 48 10/100/1000 Base-T ports and four 10 Gb/s SFP+ ports.
pprx51052GPX	{ boards 374 }	AT-x510-52GPX board with 48 10/100/1000 Base-T ports, four 10 Gb/s SFP+ ports and PSE function available on pins 1/2 and 3/6 (Mode A) of every copper port.

Table 64-5: Object identifiers for interface types

Object	Object Identifier	Description
iftypes	{ objects 3 }	
ifaceEth	{ iftypes 1 }	Ethernet
ifaceSyn	{ iftypes 2 }	Synchronous
ifaceAsyn	{ iftypes 3 }	Asynchronous
ifaceBri	{ iftypes 4 }	BRI ISDN
ifacePri	{ iftypes 5 }	PRI ISDN
ifacePots	{ iftypes 6 }	POTS (voice)
ifaceGBIC	{ iftypes 7 }	GBIC (Gigabit Interface Converter)

Table 64-6: Object identifiers for chip sets

Object	**Object Identifier**	**Description**
chips	{ objects 4 }	
chip68020Cpu	{ chips 1 }	MC68020 CPU
chip68340Cpu	{ chips 2 }	MC68340 CPU
chip68302Cpu	{ chips 3 }	MC68302 CPU
chip68360Cpu	{ chips 4 }	MC68360 CPU
chip860TCpu	{ chips 5 }	MPC860T CPU
chipRtc1	{ chips 21 }	Real Time Clock v1
chipRtc2	{ chips 22 }	Real Time Clock v2
chipRtc3	{ chips 23 }	Real Time Clock v3
chipRtc4	{ chips 24 }	Real Time Clock v4
chipRam1mb	{ chips 31 }	1 MB RAM
chipRam2mb	{ chips 32 }	2 MB RAM
chipRam3mb	{ chips 33 }	3 MB RAM
chipRam4mb	{ chips 34 }	4 MB RAM
chipRam6mb	{ chips 36 }	6 MB RAM
chipRam8mb	{ chips 38 }	8 MB RAM
chipRam12mb	{ chips 42 }	12 MB RAM
chipRam16mb	{ chips 46 }	16 MB RAM
chipRam20mb	{ chips 50 }	20 MB RAM
chipRam32mb	{ chips 62 }	32 MB RAM
chipFlash1mb	{ chips 71 }	1 MB FLASH memory
chipFlash2mb	{ chips 72 }	2 MB FLASH memory
chipFlash3mb	{ chips 73 }	3 MB FLASH memory
chipFlash4mb	{ chips 74 }	4 MB FLASH memory
chipFlash6mb	{ chips 76 }	6 MB FLASH memory
chipFlash8mb	{ chips 78 }	8 MB FLASH memory
chipPem	{ chips 120 }	Processor Enhancement Module

AT-SYSINFO-MIB

AT-SYSINFO-MIB defines objects that describe generic system information and environmental monitoring. Objects in this group have the object identifier sysinfo ({ atRouter 3 }). Table 64-7 lists the objects supported by the AlliedWare Plus^{TM} sysinfo MIB.

Table 64-7: Objects defined in AT-SYSINFO-MIB

Object	Description
sysinfo { atRouter 1 } (1.3.6.1.4.1.207.8.4.4.3)	Subtree containing generic system information.
fanAndPs {sysinfo 1 }	A collection of objects for monitoring fans and power supplies. For devices running the AlliedWare Plus^{TM} Operating System, these objects are superceded by objects in the AT-ENVMON-MIB (see "AT-ENVMONv2-MIB" on page 64.18).
restartGroup {sysinfo 2 }	A collection of objects and traps for activating and monitoring restarts. This group is not supported by devices running the AlliedWare Plus^{TM} Operating System.
cpu {sysinfo 3 }	A collection of objects containing information about the CPU utilization over different periods of time. All values are expressed as a percentage - integer in range 0 to 100.
cpuUtilisationMax {cpu 1 }	Maximum CPU utilization since the device was last restarted.
cpuUtilisationAvg {cpu 2 }	Average CPU utilization since the device was last restarted.
cpuUtilisationAvgLastMinute {cpu 3 }	Average CPU utilization over the past minute.
cpuUtilisationAvgLast10Seconds {cpu 4 }	Average CPU utilization over the past ten seconds.
cpuUtilisationAvgLastSecond {cpu 5 }	Average CPU utilization over the past second.
cpuUtilisationAvgMaxLast5Minutes {cpu 6 }	Maximum CPU utilization over the last 5 minutes.
cpuUtilisationAvgLast5Minutes {cpu 7 }	Average CPU utilization over the past 5 minutes.
cpuUtilisationStackTable {cpu 8 }	A list of stack members.
cpuUtilisationStackEntry {cpuUtilisationStackTable 1}	A set of parameters that describe the CPU utilisation of a stack member
cpuUtilisationStackId {cpuUtilisationStackEntry 1}	Stack member ID.
cpuUtilisationStackMax {cpuUtilisationStackEntry 2}	Maximum CPU utilisation since the router was last restarted. Expressed as a percentage.
cpuUtilisationStackAvg {cpuUtilisationStackEntry 3}	Average CPU utilisation since the router was last restarted. Expressed as a percentage.

Table 64-7: Objects defined in AT-SYSINFO-MIB

Object	Description
cpuUtilisationStackAvgLastMinute {cpuUtilisationStackEntry 4}	Average CPU utilisation over the past minute. Expressed as a percentage.
cpuUtilisationStackAvgLast10Seconds {cpuUtilisationStackEntry 5}	Average CPU utilisation over the past ten seconds. Expressed as a percentage.
cpuUtilisationStackAvgLastSecond {cpuUtilisationStackEntry 6}	Average CPU utilisation over the past second. Expressed as a percentage.
cpuUtilisationStackMaxLast5Minutes {cpuUtilisationStackEntry 7}	Maximum CPU utilisation over the last 5 minutes. Expressed as a percentage.
cpuUtilisationStackAvgLast5Minutes {cpuUtilisationStackEntry 8}	Average CPU utilisation over the past 5 minutes. Expressed as a percentage.
sysTemperature {sysinfo 4 }	A collection of objects and traps for monitoring and managing the temperature status. For devices running the AlliedWare Plus^{TM} Operating System.
atContactDetails {sysinfo 5 }	Contact details for Allied Telesis.
memory {sysinfo 7 }	A collection of objects and traps for monitoring memory usage and status.
atEnvMonv2 {sysinfo 12 }	AT Environment Monitoring v2 MIB for managing and reporting data relating to voltage rails, fan speeds, temperature sensors and power supply units. Objects under this portion of the OID are shown in the "AT-ENVMONv2-MIB" on page 64.18.
vcstack {sysinfo 13 }	A collection of objects for managing Virtual Chassis Stacking in AlliedWare Plus^{TM}. See "AT-VCSTACK-MIB" on page 64.24.
atPortInfo {sysinfo 14 }	Objects containing information about the transceiver of an interface. This portion of the object tree is documented separately in: "AT-PORTINFO" on page 64.15.
atVlanInfo {sysinfo 16 }	A collection of objects for counting bytes or incoming frames within a selected VLAN. Objects under this portion of the OID are shown in the "AT-VLANINFO-MIB" on page 64.30.
{sysinfo 17 } to {sysinfo 19 }	These objects are not supported on your switch.
user {sysinfo 20 }	Contains objects for displaying information of users currently logged into a device, or configured in its local database. Objects under this portion of the OID are shown in the "AT-USER-MIB" on page 64.31.
resource {sysinfo 21 }	Contains objects for displaying hardware resource information. Objects under this portion of the OID are shown in the "AT-RESOURCE-MIB" on page 64.34.
license {sysinfo 22 }	This MIB, is used for listing applied software licenses, adding new licenses, and deleting existing licenses. Objects under this portion of the OID are shown in the "AT-LICENSE-MIB" on page 64.36.

AT-PORTINFO

This table defines objects for managing interface port objects such as transceivers. Objects in this group have the object identifier vcstack ({ sysinfo 14 }), OID path, 1.3.6.1.4.1.207.8.4.4.3.14.

Table 64-8: Objects defined in AT-ATPORTINFO portion of the MIB

Object / Object Identifier	Description
atPortInfo {sysinfo 14}	This object returns information about interface transceivers.
atPortInfoTransceiverTable {atPortInfo 1}	A table of information about the transceiver of a interface.
atPortInfoTransceiverEntry {atPortInfoTransceiverTable 1}	The description, the transceiver type of a interface.
atPortInfoTransceiverifIndex {atPortInfoTransceiverEntry 1}	The ifIndex for the interface represented by this entry of the interfaces table.
atPortInfoTransceiverType {atPortInfoTransceiverEntry 1}	This object indicates the type of transceiver on a interface. It contains the following value list objects:
rj45 {atPortInfoTransceiverType 1}	RJ 45 Interface
sfp-px {atPortInfoTransceiverType 2}	SFP transceiver
sfp-bx10 {atPortInfoTransceiverType 3}	SFP transceiver
sfp-fx {atPortInfoTransceiverType 4}	SFP transceiver
sfp-100base-lx {atPortInfoTransceiverType 5}	SFP transceiver
sfp-t {atPortInfoTransceiverType 6}	SFP transceiver
sfp-cx {atPortInfoTransceiverType 7}	SFP transceiver
sfp-zx {atPortInfoTransceiverType 8}	SFP transceiver
sfp-1x {atPortInfoTransceiverType 9}	SFP transceiver
sfp-sx {atPortInfoTransceiverType 10}	SFP transceiver
sfp-oc3-1r {atPortInfoTransceiverType 11}	SFP transceiver
sfp-oc3-ir {atPortInfoTransceiverType 12}	SFP transceiver
sfp-oc3mm {atPortInfoTransceiverType 13}	SFP transceiver
xfp-srsw {atPortInfoTransceiverType 14}	XFP tranceiver

Table 64-8: Objects defined in AT-ATPORTINFO portion of the MIB(cont.)

Object / Object Identifier	Description
xfp-1r-1w {atPortInfoTransceiverType 15}	XFP tranceiver
xpf-erew {atPortInfoTransceiverType 16}	XFP tranceiver
xfp-sr {atPortInfoTransceiverType 17}	XFP tranceiver
xfp-lr {atPortInfoTransceiverType 18}	XFP tranceiver
xfp-er {atPortInfoTransceiverType 19}	XFP tranceiver
xfp-lrm {atPortInfoTransceiverType 20}	XFP tranceiver
xfp-sw {atPortInfoTransceiverType 21}	XFP tranceiver
xfp-lw {atPortInfoTransceiverType 22}	XFP tranceiver
xfp-ew {atPortInfoTransceiverType 23}	XFP tranceiver
unknown {atPortInfoTransceiverType 24}	unknown transceiver
empty {atPortInfoTransceiverType 25}	empty
sfpp-sr {atPortInfoTransceiverType 26}	SFP tranceiver
sfpp-lr {atPortInfoTransceiverType 27}	SFP tranceiver
sfpp-er {atPortInfoTransceiverType 28}	SFP tranceiver
sfpp-lrm {atPortInfoTransceiverType 29}	SFP tranceiver
inf-1-x-copper-pasv {atPortInfoTransceiverType 30}	
inf-1-x-copper-actv {atPortInfoTransceiverType 31}	
inf-1-x-lx {atPortInfoTransceiverType 32}	
inf-1-x-sx {atPortInfoTransceiverType 33}	

Table 64-8: Objects defined in AT-ATPORTINFO portion of the MIB(cont.)

Object / Object Identifier	Description
cx4 {atPortInfoTransceiverType 34}	
atPortRenumberEvents {atPortInfo 2}	The number of times that port number values (represented by the dot1dBasePort object in BRIDGE-MIB), have been re-assigned due to stack member leave/join events or XEM hot-swap events, since the system was initialised.

AT-ENVMONv2-MIB

The AT Environment Monitoring v2 MIB (atEnvMonv2-MIB) has the object path 207.8.4.4.3.12. It contains objects for managing and reporting data relating to fans, voltage rails, temperature sensors and power supply units installed in the device (Table 64-9). Objects in this group have the object identifier EnvMonv2 ({ sysinfo 12 }).

Table 64-9: Objects defined in AT-ENVMONV2-MIB

Object / Object Identifier	**Description**
atEnvMonv2Notifications { atEnvMonv2 0 } Or (207.8.4.4.3.12.0)	A collection of traps (notification) objects for monitoring fans, voltage rails, temperature sensors, and power supply bays.
atEnvMonv2FanAlarmSetNotify { atEnvMonv2Notifications 1 }	A notification that is generated when the monitored speed of a fan drops below its lower threshold. It returns the value of: 1. atEnvMonv2FanStackMemberId 2. atEnvMonv2FanBoardIndex 3. atEnvMonv2FanIndex 4. atEnvMonv2FanDescription 5. atEnvMonv2FanLowerThreshold 6. atEnvMonv2FanCurrentSpeed
atEnvMonv2FanAlarmClearedNotify { atEnvMonv2Notifications 2 }	Notification generated when the monitored speed of a fan returns to an acceptable value, the fan having previously been in an alarm condition. It returns the value of: 7. atEnvMonv2FanStackMemberId 8. atEnvMonv2FanBoardIndex 9. atEnvMonv2FanIndex 10. atEnvMonv2FanDescription 11. atEnvMonv2FanLowerThreshold 12. atEnvMonv2FanCurrentSpeed
atEnvMonv2VoltAlarmSetNotify { atEnvMonv2Notifications 3 }	Notification generated when the voltage of a monitored voltage rail, goes out of tolerance by either dropping below its lower threshold, or exceeding its upper threshold. It returns the value of: 1. atEnvMonv2VoltageStackMemberId 2. atEnvMonv2VoltageBoardIndex 3. atEnvMonv2VoltageIndex 4. atEnvMonv2VoltageDescription 5. atEnvMonv2VoltageUpperThreshold 6. atEnvMonv2VoltageLowerThreshold 7. atEnvMonv2VoltageCurrent (i.e. the voltage currently being measured).
atEnvMonv2VoltAlarmClearedNotify { atEnvMonv2Notifications 4 }	Notification generated when the voltage of a monitored voltage rail returns to an acceptable value, having previously been in an alarm condition. It returns the value of: 1. atEnvMonv2VoltageStackMemberId 2. atEnvMonv2VoltageBoardIndex 3. atEnvMonv2VoltageIndex 4. atEnvMonv2VoltageDescription 5. atEnvMonv2VoltageUpperThreshold 6. atEnvMonv2VoltageLowerThreshold 7. atEnvMonv2VoltageCurrent (i.e. the voltage currently being measured).

Table 64-9: Objects defined in AT-ENVMONV2-MIB(cont.)

Object / Object Identifier	Description
atEnvMonv2TempAlarmSetNotify { atEnvMonv2Notifications 5 }	Notification generated when a monitored temperature exceeds its upper threshold. It returns the value of: 1. atEnvMonv2TemperatureStackMemberId 2. atEnvMonv2TemperatureBoardIndex 3. atEnvMonv2TemperatureIndex 4. atEnvMonv2TemperatureDescription 5. atEnvMonv2TemperatureUpperThreshold 6. atEnvMonv2TemperatureCurrent
atEnvMonv2TempAlarmClearedNotify { atEnvMonv2Notifications 6 }	Notification generated when a monitored temperature returns to an acceptable value, having previously been in an alarm condition. It returns the value of: 1. atEnvMonv2TemperatureStackMemberId 2. atEnvMonv2TemperatureBoardIndex 3. atEnvMonv2TemperatureIndex 4. atEnvMonv2TemperatureDescription 5. atEnvMonv2TemperatureUpperThreshold
atEnvMonv2PsbAlarmSetNotify { atEnvMonv2Notifications 7 }	Notification generated when a monitored parameter of a power supply bay device goes out of tolerance. It returns the value of: 1. atEnvMonv2PsbSensorStackMemberId 2. atEnvMonv2PsbSensorBoardIndex 3. atEnvMonv2PsbSensorIndex 4. atEnvMonv2PsbSensorType 5. atEnvMonv2PsbSensorDescription
atEnvMonv2PsbAlarmClearedNotify { atEnvMonv2Notifications 8 }	Notification generated when a monitored parameter of a power supply bay device returns to an acceptable value, having previously been in an alarm condition. It returns the value of: 1. atEnvMonv2PsbSensorStackMemberId 2. atEnvMonv2PsbSensorBoardIndex 3. atEnvMonv2PsbSensorIndex 4. atEnvMonv2PsbSensorType 5. atEnvMonv2PsbSensorDescription
atEnvMonv2FanTable { EnvMonv2 1 }	Table of information about fans installed in the device that have their fan speeds monitored by environment monitoring hardware, indexed by: 1. atEnvMonv2FanStackMemberId 2. atEnvMonv2FanBoardIndex 3. atEnvMonv2FanIndex
atEnvMonv2FanEntry { atEnvMonv2FanTable 1 }	Description, current speed, lower threshold speed and current status of a single fan.
atEnvMonv2FanStackMemberId { atEnvMonv2FanEntry 1 }	Index of the stack member hosting this fan.
atEnvMonv2FanBoardIndex { atEnvMonv2FanEntry 2 }	Index of the board hosting this fan in the board table.
atEnvMonv2FanIndex { atEnvMonv2FanEntry 3 }	Numeric identifier of this fan on its host board.
atEnvMonv2FanDescription { atEnvMonv2FanEntry 4 }	Description of this fan.

Table 64-9: Objects defined in AT-ENVMONV2-MIB(cont.)

Object / Object Identifier	Description
atEnvMonv2FanCurrentSpeed { atEnvMonv2FanEntry 5 }	Current speed of this fan in revolutions per minute.
atEnvMonv2FanLowerThreshold { atEnvMonv2FanEntry 6 }	Minimum acceptable speed of the fan in revolutions per minute.
atEnvMonv2FanStatus { atEnvMonv2FanEntry 7 }	Whether this fan is currently in an alarm condition. The values can be: 1. Failed. Means that the current speed is too low. 2. Good. Means that the current speed is acceptable.
atEnvMonv2VoltageTable { atEnvMonv2 2 }	Table of information about voltage rails in the device that are monitored by environment monitoring hardware, indexed by: 1. atEnvMonv2VoltageStackMemberId ■ atEnvMonv2VoltageBoardIndex ■ atEnvMonv2VoltageIndex
atEnvMonv2VoltageEntry { atEnvMonv2VoltageTable 1 }	Description, current value, upper & lower threshold settings and current status of a single voltage rail.
atEnvMonv2VoltageStackMemberId { atEnvMonv2VoltageEntry 1 }	Index of the stack member hosting this voltage sensor.
atEnvMonv2VoltageBoardIndex { atEnvMonv2VoltageEntry 2 }	Index of the board hosting this voltage sensor in the board table.
atEnvMonv2VoltageIndex { atEnvMonv2VoltageEntry 3 }	Numeric identifier of this voltage rail on its host board.
atEnvMonv2VoltageDescription { atEnvMonv2VoltageEntry 4 }	Description of this voltage rail.
atEnvMonv2VoltageCurrent { atEnvMonv2VoltageEntry 5 }	Current reading of this voltage rail in millivolts.
atEnvMonv2VoltageUpperThreshold { atEnvMonv2VoltageEntry 6 }	Maximum acceptable reading of this voltage rail in millivolts.
atEnvMonv2VoltageLowerThreshold { atEnvMonv2VoltageEntry 7 }	Minimum acceptable reading of this voltage rail in millivolts.
atEnvMonv2VoltageStatus { atEnvMonv2VoltageEntry 8 }	Whether this voltage rail is currently in an alarm condition. Possible values are: 1. outOfRange (1) - means that the current reading is outside the threshold range. 2. inRange (2) - means that the current reading is acceptable.
atEnvMonv2TemperatureTable { atEnvMonv2 3 }	Table of information about temperature sensors in the device that are monitored by environment monitoring hardware, indexed by: 1. atEnvMonv2TemperatureStackMemberId 2. atEnvMonv2TemperatureBoardIndex 3. atEnvMonv2TemperatureIndex 4. atEnvMonv2TemperatureDescription 5. atEnvMonv2TemperatureCurrent 6. atEnvMonv2TemperatureUpperThreshold 7. atEnvMonv2TemperatureStatus
atEnvMonv2TemperatureEntry { atEnvMonv2TemperatureTable 1 }	Description, current value, upper threshold setting and current status of a single temperature sensor.
atEnvMonv2TemperatureStackMemberId { atEnvMonv2TemperatureEntry 1 }	Index of the stack member hosting this temperature sensor.

Table 64-9: Objects defined in AT-ENVMONV2-MIB(cont.)

Object / Object Identifier	Description
atEnvMonv2TemperatureBoardIndex { atEnvMonv2TemperatureEntry 2 }	Index of the board hosting this temperature sensor in the board table.
atEnvMonv2TemperatureIndex { atEnvMonv2TemperatureEntry 3 }	Numeric identifier of this temperature sensor on its host board.
atEnvMonv2TemperatureDescription { atEnvMonv2TemperatureEntry 4 }	Description of this temperature sensor.
atEnvMonv2TemperatureCurrent { atEnvMonv2TemperatureEntry 5 }	Current reading of this temperature sensor in degrees Celsius.
atEnvMonv2TemperatureUpperThreshold { atEnvMonv2TemperatureEntry 6 }	Maximum acceptable reading for this temperature sensor in degrees Celsius.
atEnvMonv2TemperatureStatus { atEnvMonv2TemperatureEntry 7 }	Whether this temperature sensor is currently in an alarm condition. Can be: 1. outOfRange (1) - means that the current reading is outside the threshold range. 2. inRange (2) - means that the current reading is acceptable.
atEnvMonv2PsbObjects { atEnvMonv2 4 }	Collection of objects for monitoring power supply bays in the system and any devices that are installed. It contains the following objects: 1. atEnvMonv2PsbTable ■ atEnvMonv2PsbSensorTable
atEnvMonv2PsbTable { atEnvMonv2PsbObjects 1 }	Table of information about power supply bays in the system, indexed by: 1. atEnvMonv2PsbHostStackMemberId 2. atEnvMonv2PsbHostBoardIndex 3. atEnvMonv2PsbHostSlotIndex 4. atEnvMonv2PsbHeldBoardIndex 5. atEnvMonv2PsbHeldBoardId 6. atEnvMonv2PsbDescription
atEnvMonv2PsbEntry { atEnvMonv2PsbTable 1 }	Description and current status of a single power supply bay device.
atEnvMonv2PsbHostStackMemberId { atEnvMonv2PsbEntry 1 }	Index of the stack member hosting this power supply bay.
atEnvMonv2PsbHostBoardIndex { atEnvMonv2PsbEntry 2 }	Index of the board hosting this power supply bay in the board table.
atEnvMonv2PsbHostSlotIndex { atEnvMonv2PsbEntry 3 }	Index of this power supply bay slot on its host board. This index is fixed for each slot, on each type of board.
atEnvMonv2PsbHeldBoardIndex { atEnvMonv2PsbEntry 4 }	Index of a board installed in this power supply bay. This value corresponds to atEnvMonv2PsbSensorBoardIndex for each sensor on this board. A value of 0 indicates that a board is either not present or not supported.
atEnvMonv2PsbHeldBoardId { atEnvMonv2PsbEntry 5 }	Type of board installed in this power supply bay. The values of this object are taken from the pprXxx object IDs under the boards sub-tree in the parent MIB. A value of 0 indicates that a board is either not present or not supported.

Table 64-9: Objects defined in AT-ENVMONV2-MIB(cont.)

Object / Object Identifier	Description
atEnvMonv2PsbDescription { atEnvMonv2PsbEntry 6 }	Description of this power supply bay.
atEnvMonv2PsbSensorTable { atEnvMonv2PsbObjects 2 }	Table of information about environment monitoring sensors on devices installed in power supply bays, indexed by: 1. atEnvMonv2PsbSensorStackMemberId ■ atEnvMonv2PsbSensorBoardIndex ■ atEnvMonv2PsbSensorIndex
atEnvMonv2PsbSensorEntry { atEnvMonv2PsbSensorTable 1 }	Description and current status of the sensor on a device installed in a power supply bay.
atEnvMonv2PsbSensorStackMemberId { atEnvMonv2PsbSensorEntry 1 }	Index of the stack member hosting this sensor.
atEnvMonv2PsbSensorBoardIndex { atEnvMonv2PsbSensorEntry 2 }	Index of the board hosting this sensor in the board table.
atEnvMonv2PsbSensorIndex { atEnvMonv2PsbSensorEntry 3 }	Index of this power supply bay environmental sensor on its host board.
atEnvMonv2PsbSensorType { atEnvMonv2PsbSensorEntry 4 }	Type of environmental variable this sensor detects. One of: 1. psbSensorTypeInvalid(0) ■ fanSpeedDiscrete(1) ■ temperatureDiscrete(2) ■ voltageDiscrete(3)
atEnvMonv2PsbSensorDescription { atEnvMonv2PsbSensorEntry 5 }	Description of this power supply bay environmental sensor.
atEnvMonv2PsbSensorStatus { atEnvMonv2PsbSensorEntry 6 }	Whether this environmental sensor is currently in an alarm condition. One of: 1. failed (1) - the device is in a failure condition 2. good (2) - the device is functioning normally. 3. notPowered (3) - a PSU is installed, but not powered up
atEnvMonv2PsbSensorReading { atEnvMonv2PsbSensorEntry 7 }	An indication of whether this environmental sensor is currently reading a value for the monitored device. A value of 'no' indicates that there is no current reading, 'yes' indicates that the monitored device is supplying a reading. 1. no 2. yes
atEnvMonv2Traps { atEnvMonv2 5 } (207.8.4.4.3.12.5)	Note that ojects under this portion of the tree have been deprecated, and replaced by objects under the tree portion 207.8.4.4.3.12.0.
atEnvMonv2FaultLedTable { atEnvMonv2 6 }	Table detailing any LED fault indications on the device, indexed by: 1. atEnvMonv2FaultLedStackMemberId
atEnvMonv2FaultLedEntry { atEnvMonv2FaultLedTable 1 }	Information pertaining to a given fault LED.
atEnvMonv2FaultLedStackMemberId { atEnvMonv2FaultLedEntry 1 }	Index of the stack member hosting this fault LED.
atEnvMonv2FaultLed1Flash { atEnvMonv2FaultLedEntry 2 }	Indicates whether a fault LED is currently showing a system failure by flashing once. Values can be: 1. heatsinkFanFailure (1) - indicates that one or more heatsink fans have failed, or are operating below the recommended speed ■ noFault (2)

Table 64-9: Objects defined in AT-ENVMONV2-MIB(cont.)

Object / Object Identifier	Description
atEnvMonv2FaultLed2Flashes { atEnvMonv2FaultLedEntry 3 }	Indicates whether a fault LED is currently showing a system failure by flashing twice. Values can be: 1. chassisFanFailure (1) - indicates that one or both of the chassis fans are not installed, or the fans are operating below the recommended speed 2. noFault (2)
atEnvMonv2FaultLed3Flashes { atEnvMonv2FaultLedEntry 4 }	Indicates whether a fault LED is currently showing a system failure by flashing three times. Values can be: 1. sensorFailure (1) - indicates that the ability to monitor temperature or fans has failed 2. noFault (2)
atEnvMonv2FaultLed4Flashes { atEnvMonv2FaultLedEntry 5 }	Indicates whether a fault LED is currently showing a system failure by flashing four times. Values can be: 1. xemInitialisationFailure (1) - indicates that a XEM failed to initialise or is incompatible 2. noFault (2)
atEnvMonv2FaultLed5Flashes { atEnvMonv2FaultLedEntry 6 }	Indicates whether a fault LED is currently showing a system failure by flashing five times. This flashing sequence is not currently in use. Value is: ■ noFault (2)
atEnvMonv2FaultLed6Flashes { atEnvMonv2FaultLedEntry 7 }	Indicates whether a fault LED is currently showing a system failure by flashing six times. Values can be: 1. temperatureFailure (1) - indicates that the device's temperature has exceeded the recommended threshold 2. noFault (2)

AT-VCSTACK-MIB

AT-VCSTACK-MIB defines objects for managing Virtual Chassis Stacking (Table 64-2). Objects in this group have the object identifier vcstack ({ sysinfo 13 }), OID path, 1.3.6.1.4.1.207.8.4.4.3.13.

Figure 64-2 on page 64.24 shows the tree structure of the AT-VCSTACK objects.

Figure 64-2: The AT-VCSTACK MIB sub-tree

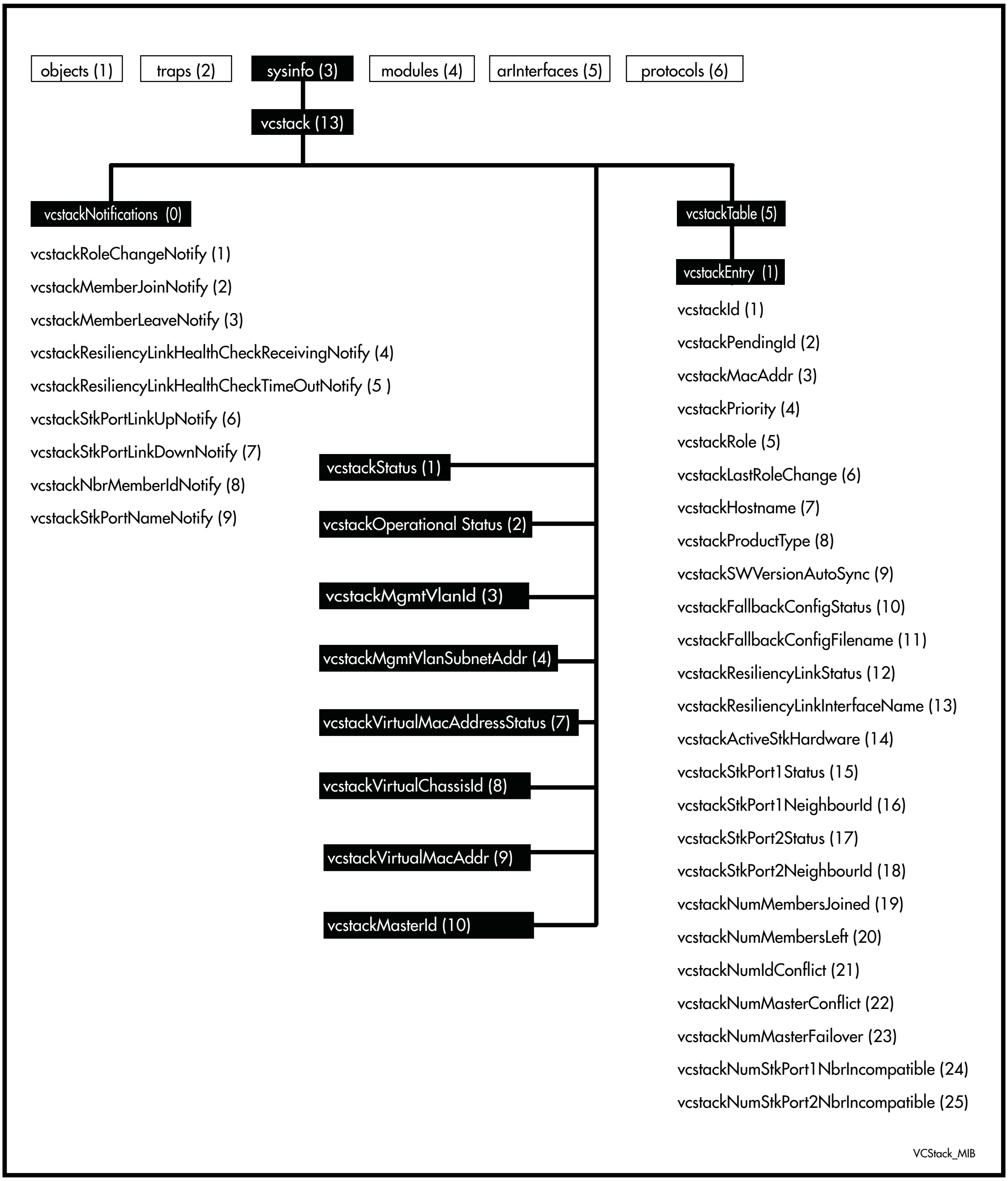

objects (1)
traps (2)
sysinfo (3)
modules (4)
arInterfaces (5)
protocols (6)
vcstack (13)
vcstackNotifications (0)
vcstackRoleChangeNotify (1)
vcstackMemberJoinNotify (2)
vcstackMemberLeaveNotify (3)
vcstackResiliencyLinkHealthCheckReceivingNotify (4)
vcstackResiliencyLinkHealthCheckTimeOutNotify (5)
vcstackStkPortLinkUpNotify (6)
vcstackStkPortLinkDownNotify (7)
vcstackNbrMemberIdNotify (8)
vcstackStkPortNameNotify (9)
vcstackStatus (1)
vcstackOperational Status (2)
vcstackMgmtVlanId (3)
vcstackMgmtVlanSubnetAddr (4)
vcstackVirtualMacAddressStatus (7)
vcstackVirtualChassisId (8)
vcstackVirtualMacAddr (9)
vcstackMasterId (10)
vcstackTable (5)
vcstackEntry (1)
vcstackId (1)
vcstackPendingId (2)
vcstackMacAddr (3)
vcstackPriority (4)
vcstackRole (5)
vcstackLastRoleChange (6)
vcstackHostname (7)
vcstackProductType (8)
vcstackSWVersionAutoSync (9)
vcstackFallbackConfigStatus (10)
vcstackFallbackConfigFilename (11)
vcstackResiliencyLinkStatus (12)
vcstackResiliencyLinkInterfaceName (13)
vcstackActiveStkHardware (14)
vcstackStkPort1Status (15)
vcstackStkPort1NeighbourId (16)
vcstackStkPort2Status (17)
vcstackStkPort2NeighbourId (18)
vcstackNumMembersJoined (19)
vcstackNumMembersLeft (20)
vcstackNumIdConflict (21)
vcstackNumMasterConflict (22)
vcstackNumMasterFailover (23)
vcstackNumStkPort1NbrIncompatible (24)
vcstackNumStkPort2NbrIncompatible (25)
VCStack_MIB

Table 64-10: Objects defined in AT-VCSTACK-MIB

Object	Description
vcstack OID { sysinfo (13) } (207.8.4.4.3.13)	Overall stack status.
vcstackNotifications { sysinfo (13) }	Overall stack status.
vcstackRoleChangeNotify { vcstackNotifications 1 }	The stack status can take one of the following states: 1. vcstackId 2. vcstackRole
vcstackMemberJoinNotify { vcstackNotifications 2 }	Notification generated when a member joins the stack. Displays the objects: 1. vcstackId 2. vcstackNbrMemberIdNotify
vcstackMemberLeaveNotify { vcstackNotifications 3 }	Notification generated when a member leaves the stack. Displays the objects: 1. vcstackId 2. vcstackNbrMemberIdNotify
vcstackResiliencyLinkHealthCheckReceivingNotify { vcstackNotifications 4 }	Notification generated when the resiliency link is activated. Displays the objects: 1. vcstackId 2. vcstackResiliencyLinkInterfaceName
vcstackResiliencyLinkHealthCheckTimeOutNotify { vcstackNotifications 5 }	Notification generated when the member's receive timer has timed-out, indicating that the member has lost contact with the Master via the resiliency link. Displays the objects: 1. vcstackId 2. vcstackResiliencyLinkInterfaceName
vcstackStkPortLinkUpNotify { vcstackNotifications 6 }	Notification generated when the stack port link is up. Displays the objects: 1. vcstackId 2. vcstackStkPortNameNotify
vcstackStkPortLinkDownNotify { vcstackNotifications 7 }	Notification generated when the stack port link is down. Displays the objects: 1. vcstackId 2. vcstackStkPortName
vvcstackNbrMemberIdNotify { vcstackNotifications 8 }	The stack member id related to this notification
vcstackStkPortNameNotify { vcstackNotifications 9 }	The stack port name related to this notification
vcstackStatus { vcstack 1 }	The overall stack status. 1. normalOperation(1) 2. operatingInFailoverState(2) 3. standaloneUnit(3) 4. ringTopologyBroken(4)

Table 64-10: Objects defined in AT-VCSTACK-MIB(cont.)

Object	Description
vcstackOperationalStatus { vcstack 2 }	The operational status of the stack can be either: 1. enabled (1) 2. disabled (2)
vcstackMgmtVlanId	The current stacking management VLAN ID
vcstackMgmtVlanSubnetAddr { vcstack 3 }	The current stacking management VLAN subnet address
vcstackTable { vcstack 4 }	Table of information about stack members, indexed by vcstackId.
vcstackEntry { vcstackTable 1 }	Information about a single stack member, indexed by vcstackId.
vcstackId { vcstackEntry 1 }	Stack member ID.
vcstackPendingId { vcstackEntry 2 }	Pending stack member ID.
vcstackMacAddr { vcstackEntry 3 }	Stack member's hardware MAC address.
vcstackPriority { vcstackEntry 4 }	Priority for election of the stack master. The lowest number has the highest priority.
vcstackRole { vcstackEntry 5 }	Stack member's role in the stack. Can be one of the following: 1. leaving (1) 2. discovering (2) 3. synchronizing (3) 4. backupMember (4) 5. pendingMaster (5) 6. disabledMaster (6) 7. fallbackMaster (7) 8. activeMaster (8)
vcstackLastRoleChange { vcstackEntry 6}	Time and date when the stack member last changed its role in the stack.
vcstackHostname { vcstackEntry 7 }	Stack member's hostname.
vcstackProductType { vcstackEntry 8 }	Stack members product type.
vcstackSWVersionAutoSync { vcstackEntry 9}	Whether or not the stack member's software is automatically upgraded.
vcstackFallbackConfigStatus { vcstackEntry 10 }	Status of the fallback configuration file. Can be one of: 1. fileExists (1) 2. fileNotFound (2) 3. notConfigured (3)
vcstackFallbackConfigFilename { vcstackEntry 11}	Filename of the fallback configuration file.

Table 64-10: Objects defined in AT-VCSTACK-MIB(cont.)

Object		Description
	vcstackResiliencyLinkStatus { vcstackEntry 12 }	Status of the stack members resiliency link. Can be one of: 1. configured (1) 2. successful (2) 3. failed (3) 4. notConfigured (4)
	vcstackResiliencyLinkInterfaceName { vcstackEntry 13 }	Name of the interface the resiliency link is configured on.
	vcstackActiveStkHardware { vcstackEntry 14 }	Stack ports hardware type. Can be one of: 0. value (0) is now obsolete 1. xemStk (1) 2. builtinStackingPorts (2) 3. none (3) is now obsolete 4. stackXG (4)
	vcstackStkPort1Status { vcstackEntry 15 }	Status of stack-port 1. Can be one of the following: 1. down (1) 2. neighbourIncompatible (2) 3. discoveringNeighbour (3) 4. learntNeighbour (4)
	vcstackStkPort1NeighbourId { vcstackEntry 16 }	ID of the neighbor on stack-port 1. Zero indicates no learned neighbor.
	vcstackStkPort2Status { vcstackEntry 17 }	Status of stack-port 2. Can be one of: 1. down (1) 2. neighbourIncompatible (2) 3. discoveringNeighbour (3) 4. learntNeighbour (4)
	vcstackStkPort2NeighbourId { vcstackEntry 18 }	ID of the neighbor on stack-port 2. Zero indicates no learned neighbor.
	vcstackNumMembersJoined { vcstackEntry 19 }	Number of times the stack has acquired a member.
	vcstackNumMembersLeft { vcstackEntry 20 }	Number of times the stack has lost a member.
	vcstackNumIdConflict { vcstackEntry 21 }	Number of times that a stack member ID conflict has occurred.
	vcstackNumMasterConflict { vcstackEntry 22}	Number of times that a stack master conflict has occurred.
	vcstackNumMasterFailover { vcstackEntry 23 }	Number of times that the stack master has failed.
	vcstackNumStkPort1NbrIncompatible { vcstackEntry 24 }	Number of times that the neighbor on stack port 1 was incompatible.
	vcstackNumStkPort2NbrIncompatible { vcstackEntry 25 }	Number of times that the neighbor on stack port 2 was incompatible.

Table 64-10: Objects defined in AT-VCSTACK-MIB(cont.)

Object	Description
vcstackReadinessStatus { vcstackEntry 26 }	Indicates how readily a stack member can take over as master if the current stack master were to fail. Possible values are: 1. init (1) - the stack member is completing startup initialization. 2. syncing (2) - the stack member is synchronizing state information with the stack master following startup. 3. ready (3) - the stack member is fully synchronized with the current master and is ready to take over immediately. 4. syncError (4) - state information on the stack member is not correctly Note that for a stack member to take over as stack master with the least possible network disruption, it must have the 'ready (3)' status.
vcstackTraps {vcstack 6}	**Note** Traps under this portion of the object tree have have been deprecated pending obsoletion: 1. vcstackRoleChange 2. vcstackMemberJoin 3. vcstackMemberLeave 4. vcstackResiliencyLinkHealthCheckReceiving 5. vcstackResiliencyLinkHealthCheckTimeOut 6. vcstackStkPortLinkUp 7. vcstackStkPortLinkDown 8. vcstackNbrMemberId 9. vcstackStkPortName
vcstackVirtualMacAddressStatus {vcstack 7}	Indicates whether the virtual MAC address is enabled or disabled. Read-only object. Values are: 1. enabled(1) 2. disabled(2)
vcstackVirtualChassisId {vcstack 8}	Displays the current virtual chassis ID. Read-only object.
vcstackVirtualMacAddr {vcstack 9}	Displays the virtual MAC address used by the stack. Read-only object.
vcstackMasterId {vcstack 10}	Displays the stack ID of the master unit, or the stack ID of the standalone unit. Value range 1-8. Read-only object
vcstackDisabledMasterMonitoringStatus {vcstack 11}	Whether the disabled master monitoring function is enabled or disabled.Values are: 1. enabled(1) 2. disabled(2) 3. inactive(3)

AT-MIBVERSION-MIB

The AT-MIBVERSION-MIB contains an object to display the last software release that contained changes to the supported AT Enterprise MIB definition files (Table). Objects in this group have the object identifier atMibsetVersion ({ sysinfo 15 }).

Object defined in AT-MIBVERSION-MIB

Object	Object Identifier	Description
atMibsetVersion	{ sysinfo 15 }	This object returns a five digit integer which indicates the last software release that contained changes to the supported AT Enterprise MIB definition files. For example, If the currently loaded software release on the device is 5.3.1-0.3 but the Enterprise MIBs have not changed since 5.3.1-0.1, then the value returned will be 53101.

AT-VLANINFO-MIB

AT-SMI-MIB defines objects relating to VLANs. These objects are imported by other component MIBs of the Allied Telesis Enterprise MIB. Objects in this group have the object identifier atMibsetVersion ({ sysinfo 16 }).

Table 64-11: AT Enterprise MIB - High Level Structure

Object	Description
vlaninfo {sysinfo 16}	Root of the Allied Telesis Enterprise MIB under the private(4) node defined in RFC1155-SMI.
atVlanStatistics {vlaninfo 1}	The number of unique VLAN statistic gathering instances defined on the device.
atVlanStatNumCollections {atVlanStatistics 1}	The number of unique VLAN statistic gathering instances defined on the device.
atVlanStatCollectionTable {atVlanStatistics 2}	A table of VLAN statistic instances.
atVlanStatCollectionEntry {atVlanStatCollectionTable 1}	Each entry represents a unique VLAN statistic gathering instance defined on the device. Sequences are:
atVlanStatCollectionName {atVlanStatCollectionEntry 1}	The name of a VLAN statistics collection instance.
atVlanStatCollectionVlanId {atVlanStatCollectionEntry 2}	The VLAN ID of ingress packets being monitored by this VLAN statistics collection instance.
atVlanStatCollectionPortMap {atVlanStatCollectionEntry 3}	A bitwise port map indicating the switch ports being monitored by this VLAN statistics collection instance. The bit position within the string, maps to the port with the same index in dot1dBasePortTable in BRIDGE-MIB. A binary '1' indicates that the port is being monitored by this VLAN statistics collection instance, with a '0' indicating that it is not.
atVlanStatCollectionIngressPkts {atVlanStatCollectionEntry 4}	The number of ingress packets received and counted by this VLAN statistics collection instance.
atVlanStatCollectionIngressOctets {atVlanStatCollectionEntry 5}	The number of octets of data received from ingress packets counted by this VLAN statistics collection instance.
atVlanStatCollectionResetStats {atVlanStatCollectionEntry 6}	When read, this object will always return 2 (false). Setting its value to 1 (true) will cause the matching VLAN statistics collection instance's ingress packets and ingress octet values to be reset to zero.

AT-USER-MIB

The AT-USER-MIB contains objects for displaying information about users currently logged into a device, or configured in the Local User Database of the device (Table 64-12). Objects in this group have the object identifier user ({ sysinfo 20 }).

Table 64-12: Objects defined in AT-USER-MIB

Object	Object Identifier	Description
userInfoTable (207.8.4.4.3.20.1)	{ user 1 }	Table containing information about users. Each entry in the table represents a user currently logged into the device. Indexed by: rscBoardType and rscBoardIndex.
userInfoEntry	{ userInfoTable 1 }	Information about a single user logged into the device.
userInfoType	{ userInfoEntry 1 }	The type of connection through which the user logged into the device. Can be: 1. console (1) 2. aux (2) 3. telnet (3) 4. script (4) 5. stack (5)
userInfoIndex	{ userInfoEntry 2 }	Index of the line upon which the user logged into the device. Can be a value in range 1 to 16.
userInfoName	{ userInfoEntry 3 }	User name of the user logged into the device.
userInfoPrivilegeLevel	{ userInfoEntry 4 }	The user's privilege level. Can be a value in range 1 to 15.
userInfoIdleTime	{ userInfoEntry 5 }	The amount of time since the user was last active, in the form hh:mm:ss.
userInfoLocation	{ userInfoEntry 6 }	The user location or login method. It can be an IP Address used by the user to telnet into the device, or an asyn port, etc.
userInfoPasswordLifetime	{ userInfoEntry 7 }	The number of days remaining until the user's password expires. Depending on the current user setting it will display one of the following: ■ No Expiry - the password will never expire (default setting) ■ x days - where **x** is the remaining lifetime of the current password (maximum lifetime value is 1000 days) ■ -x days (expired) - indicating that the current password expired **x** days ago
userInfoPasswordLastChange	{ userInfoEntry 8 }	The number of days since the password was last altered.

Table 64-12: Objects defined in AT-USER-MIB(cont.)

Object	Object Identifier	Description
userConfigTable	{ user 2 } or (207.8.4.4.3.20.2)	Table containing user configuration information. Each entry in the table relates to a user configured in the Local User Database of the device. Indexed by userConfigIndex.
userConfigEntry	{ userConfigTable 1 }	Information about a single user configured in the Local User Database of the device.
userConfigIndex	{ userConfigEntry 1 }	Unique number used to identify entries in the userConfigTable.
userConfigName	{ userConfigEntry 2 }	The user's name.
userConfigPrivilegeLevel	{ userConfigEntry 3 }	The privilege level granted to the user. Can be a value in range 1 to 15.

Table 64-12: Objects defined in AT-USER-MIB(cont.)

Object	Object Identifier	Description
userSecurityPasswordRules	{ user 3 } or (207.8.4.4.3.20.3)	Information about user password security rules.
userSecurityPasswordHistory	{ userSecurityPasswordRules 1 }	The number of previous passwords that are retained for comparison when a user password is created. A new password must be unique when compared against the previous history. A value of 0 represents no restriction. The maximum number of retained passwords is 15.
userSecurityPasswordLifetime	{ userSecurityPasswordRules 2 }	The maximum number of days that the password may persist before a change is required. A value of 0 represents no expiry. The maximum value is 1000.
userSecurityPasswordWarning	{ userSecurityPasswordRules 3 }	The number of days before the password expires that a warning message is displayed when the user logs in. A value of 0 indicates no warning. The maximum value is 1000 but must always be less than the password lifetime.
userSecurityPasswordMinLength	{ userSecurityPasswordRules 4 }	The minimum allowable password length.
userSecurityPasswordMinCategory	{ userSecurityPasswordRules 5 }	The minimum number of different categories that the password must satisfy to be considered valid. Categories are split into four groups: ■ upper-case letters ■ lower-case letters ■ digits ■ special symbols. ASCII characters not included in the previous three categories.
userSecurityPasswordForced	{ userSecurityPasswordRules 6 }	Whether or not a user with an expired password is forced to change their password at the next login. At login a user with an expired password is prompted to change their password. If the new password meets the current security password rules the user is allowed to login, otherwise they are rejected.
userSecurityPasswordReject	{ userSecurityPasswordRules 7 }	Whether or not a user login attempt with an expired password is rejected. If the user is not rejected then they can login.

AT-RESOURCE-MIB

The AT-RESOURCE-MIB contains objects for displaying system hardware resource and host information (Table 64-13). Objects in this group have the object identifier rsc ({ sysinfo 21 }), or (207.8.4.4.3.21).

Table 64-13: Objects defined in AT-RESOURCE-MIB

Object and OID	Description
resource { sysinfo 21 }	Contains objects for displaying system hardware resource and host information.
rscBoardTable { resource 1}	Table containing information about boards installed in a device. Indexed by: 1. rscStkId 2. rscResourceId
rscBoardEntry { rscBoardTable 1 }	Information about a single board installed in the device.
rscStkId { rscBoardEntry 1 }	The ID of the stack member. It is a number from 1 to 8, assigned to a stackable unit by the operating system when it is stacked. A default of 1 is given to a stand-alone unit.
rscResourceId { rscBoardEntry 2 }	The resource ID number of the board. It is a number assigned to a hardware resource when the operating system detects its existence. Can be a value in range 1 to 4294967294.
rscBoardType { rscBoardEntry 3 }	The type of board. Can be one of the following: 1. Base 2. Expansion 3. Fan module 4. PSU, etc.
rscBoardName { rscBoardEntry 4 }	The name of the board. Can be one of the following: 1. SwitchBlade x908 2. XEM-12S 3. AT-PWR05-AC, etc
rscBoardId { rscBoardEntry 5 }	The ID number of the board. Its value is an Allied Telesis assigned number, such as 274 for the XEM-12S, or 255 for the AT-9924Ts.
rscBoardBay { rscBoardEntry 6 }	The board installation location. Its value can be Bay1, Bay2, PSU1, etc. For a base board, it has a value of a single character space.
rscBoardRevision { rscBoardEntry 7 }	The revision number of the board.
rscBoardSerialNumber { rscBoardEntry 8 }	The serial number of the board.
hostInfoTable { resource 2 }	Table containing general system information. Indexed by rscStkId.
hostInfoEntry { hostInfoTable 1 }	Information about a single system parameter

Table 64-13: Objects defined in AT-RESOURCE-MIB**(cont.)**

Object and OID	Description
hostInfoDRAM { hostInfoTable 2 }	The host DRAM information.
hostInfoFlash { hostInfoTable 3 }	The host Flash information.
hostInfoUptime { hostInfoTable 4 }	The host up-time.
hostInfoBootloaderVersion { hostInfoTable 5 }	The host boot loader version.

AT-LICENSE-MIB

The AT-LICENSE-MIB contains objects for managing the AlliedWare Plus^{TM} Operating System software licenses: listing applied software licenses, adding new licenses and deleting existing licenses (Table 64-14). The objects reside in the module license { sysinfo 22 }, organized in the following groups:

- Base Software License Table - a table containing the installed base software licenses on the device
- Installed Software License Table - a list of installed software licenses; used also to remove software license from the device
- Available Software Features Table
- LicenseNew - Objects used to install a new license
- LicenseStackRemove - Objects used to remove a license across a stack of devices

Table 64-14: Objects defined in AT-LICENSE-MIB

Object	Object Identifier	Description
license	{ sysinfo 22 }	MIB containing objects for listing applied software licenses, adding new licenses, and deleting existing licenses.
baseLicenseTable	{ license 1 }	Table containing information about base software licenses installed on a device. Indexed by: ■ baseLicenseStkId
baseLicenseEntry	{ baseLicenseTable 1 }	Information about a single license installed on the device.
baseLicenseStkId	{ baseLicenseEntry 1 }	The stack member ID of the device hosting the license.
baseLicenseName	{ baseLicenseEntry 2 }	The name of the base license.
baseLicenseQuantity	{ baseLicenseEntry 3 }	The number of licenses issued for this entry.
baseLicenseType	{ baseLicenseEntry 4 }	The type of base license issued.
baseLicenseIssueDate	{ baseLicenseEntry 5 }	The date of issue of the base license.
baseLicenseExpiryDate	{ baseLicenseEntry 6 }	The expiry date of the base license.
baseLicenseFeatures	{ baseLicenseEntry 7 }	The feature set that this license enables, in the format of an octet string. Each bit in the returned octet string represents a particular feature that can be license-enabled. The bit position within the string maps to the feature entry with the same index, in licenseFeatureTable. A binary '1' indicates that the feature is included in the license; a binary '0' indicates that the feature is not included in the license.
licenseTable	{ license 2 }	Table containing information about software licenses installed on the device. Indexed by: ■ licenseStackId ■ licenseIndex
licenseEntry	{ licenseTable 1 }	Information about a single installed software license on the device.
llicenseStackId	{ licenseEntry 1 }	The stack member ID of the device hosting the license.
licenseIndex	{ licenseEntry 2 }	The index number of the license entry.
licenseName	{ licenseEntry 3 }	The name of the license.
licenseCustomer	{ licenseEntry 4 }	The name of the customer of the license.
licenseQuantity	{ licenseEntry 5 }	The number of licenses issued for this entry.
licenseType	{ licenseEntry 6 }	The type of license issued.
licenseIssueDate	{ licenseEntry 7 }	The date of issue of the license.

Table 64-14: Objects defined in AT-LICENSE-MIB(cont.)

Object	Object Identifier	Description
licenseExpiryDate	{ licenseEntry 8 }	The expiry date of the license.
licenseFeatures	{ licenseEntry 9 }	The feature set that this license enables, in the format of octet string. Each bit in the returned octet string represents a particular feature that can be license-enabled. The bit position within the string maps to the feature entry with the same index, in licenseFeatureTable. A binary '1' indicates that the feature is included in the license; a binary '0' indicates that the feature is not included in the license.
licenseRowStatus	{ licenseEntry 10 }	The current status of the license. The following values may be returned when reading this object: 1. active (1) - the license is currently installed and valid 2. notInService (2) - the license has expired or is invalid The following value may be written to this object: ■ destroy (6) - the license will be removed from the device; this may result in some features being disabled. Note that a stacked device that has a license deleted may not be able to rejoin the stack after reboot, unless the license is also deleted on all other devices in the stack.
licenseFeatureTable	{ license 3 }	Table containing all available Software Features. A feature must be license-enabled to be utilized on the device.
licenseFeatureEntry	{ licenseFeatureTable 1 }	Information about a single feature that must be license-enabled in order to be utilized on the device.
licenseFeatureIndex	{ licenseFeatureEntry 1 }	The index number of the feature which must be license-enabled.
licenseFeatureName	{ licenseFeatureEntry 2 }	The name of the feature under licensing control.
licenseFeatureStkMembers	{ licenseFeatureEntry 3 }	The set of stack members on which the feature is enabled, in the format of an octet string. Each bit in the string maps to an individual stacking member, e.g. bit one represents stacking member one, bit two represents stacking member two, etc. A bit value of '1' indicates that the applicable feature is enabled on the matching device; a bit value of '0' indicates that the feature is disabled.
licenseNew	{ license 4 }	Group of objects available for updates, used when installing a new software license on the device.
licenseNewStackId	{ licenseNew 1 }	The ID of the stacking member upon which the new license is to be installed. The value zero (0) indicates that the license should be applied to all stack members.
licenseNewName	{ licenseNew 2 }	The name of the new license to be installed.
licenseNewKey	{ licenseNew 3 }	The key for the new license to be installed.

Table 64-14: Objects defined in AT-LICENSE-MIB(cont.)

Object	Object Identifier	Description
licenseNewInstall	{ licenseNew 4 }	Used to install new licenses. Values can be: 1. true (1) 2. false (2) To commence installation, a valid license name and key must first have been set via the licenseNewName and licenseNewKey respectively. This object should then be set to the value true (1). If either the license name or key is invalid, the write operation will fail. Once installed, the software modules affected by any newly enabled features will automatically be restarted. Note that a stacked device that has a new license installed on it may not be able to rejoin the stack after reboot, unless the license is also added to all other devices in the stack. When read, the object will always return the value false (2).
licenseNewInstallStatus	{ licenseNew 5 }	The current status of the last license installation request. One of the following values is returned when reading this object: 1. idle (1) 2. processing (2) 3. success (3) 4. failed (4) When a stack license installation operation is complete the first read of this object will return either a success (3) or a failure (4) indication. Subsequent reads of this object will then return an idle (1) indication.
licenseStackRemove	{ license 5 }	Group of objects used when removing a software license across a stack of devices.
licenseStackRemoveName	{ licenseStackRemove 1 }	The name of the license to be removed from all devices across the stack, on which the license currently exists.
licenseStackRemoveExecute	{ licenseStackRemove 2 }	When set to the value true (1), the system will attempt to remove the named license from all devices across the stack on which the license currently exists. All devices in a stack must be from the same product family and the named license must activate the same feature set on all devices.
licenseStackRemoveStatus	{ licenseStackRemove 3 }	The current status of the last requested stack license removal request. One of the following values is returned when reading this object: 1. idle (1) 2. processing (2) 3. success (3) 4. failed (4) When a stack license removal operation is complete the first read of this object will return either a success (3) or failure (4) indication. Subsequent reads of this object will then return an idle (1) indication.

AT-TRIGGER-MIB

AT-TRIGGER-MIB defines objects for managing triggers (Table 64-15). Objects in this group have the object identifier trigger ({ modules 53 }). All objects in this group have read only access.

Table 64-15: Objects defined in AT-TRIGGER-MIB

Object Identifier	Description
triggerTraps { trigger 0 }	Sub-tree for all trigger traps.
triggerTrap { triggerTraps 1 }	Notification generated when a trigger is activated. It returns the value of triggerLastTriggerActivated.
triggerLastTriggerActivated { trigger 1 }	Trigger number of the most recent trigger activated on the switch.
triggerConfigInfoTable { trigger 9 }	Table of information about each trigger that has been configured, indexed by triggerNumber.
triggerConfigInfoEntry { triggerConfigInfoTable 1 }	Information about the configuration of a single trigger.
triggerNumber { triggerConfigInfoEntry 1 }	ID number of the trigger. Values are in range 1- 250.
triggerName { triggerConfigInfoEntry 2 }	Name and description of the trigger.
triggerTypeDetail { triggerConfigInfoEntry 3 }	Trigger type and its activation conditions.
triggerActiveDaysOrDate { triggerConfigInfoEntry 4 }	The days of a week or the date on which the trigger can be activated.
triggerActivateAfter { triggerConfigInfoEntry 5 }	Time after which the trigger can be activated.
triggerActivateBefore { triggerConfigInfoEntry 6 }	Time before which the trigger can be activated.
triggerActiveStatus { triggerConfigInfoEntry 7 }	Whether or not the trigger can be activated.
triggerTestMode { triggerConfigInfoEntry 8 }	Whether or not the trigger is operating in diagnostic (test) mode.
triggerSnmpTrap { triggerConfigInfoEntry 9 }	Whether or a not an SNMP trap will be generated when the trigger is activated.
triggerRepeatTimes { triggerConfigInfoEntry 10 }	Whether the trigger can repeat an unlimited number of times (continuous) or a specified number of times. If the trigger can repeat only a specified number of times, then the number of times the trigger has already been activated is displayed in brackets.
triggerLasttimeModified { triggerConfigInfoEntry 11 }	Date and time that the trigger configuration was last modified.
triggerNumberOfActivation { triggerConfigInfoEntry 12 }	Number of times the trigger has been activated since the last restart of the device.
triggerLasttimeActivation { triggerConfigInfoEntry 13 }	Date and time that the trigger was last activated.
triggerNumberOfScripts { triggerConfigInfoEntry 14 }	Number of scripts that this trigger will execute. Values are in range 0-5.
triggerScript1 { triggerConfigInfoEntry 15 }	Name of the first script that this trigger will execute if the trigger is activated.

Table 64-15: Objects defined in AT-TRIGGER-MIB(cont.)

Object Identifier	Description
triggerScript2 { triggerConfigInfoEntry 16 }	Name of the second script that this trigger will execute if the trigger is activated.
triggerScript3 { triggerConfigInfoEntry 17 }	Name of the third script that this trigger will execute if the trigger is activated.
triggerScript4 { triggerConfigInfoEntry 18 }	Name of the fourth script that this trigger will execute if the trigger is activated.
triggerScript5 { triggerConfigInfoEntry 19 }	Name of the fifth script that this trigger will execute if the trigger is activated.
triggerCounters { trigger 10 }	Collection of counters for trigger activations.
triggerNumOfActivation { triggerCounters 1 }	Number of times a trigger has been activated.
triggerNumOfActivationToday { triggerCounters 2 }	Number of times a trigger has been activated today.
triggerNumOfPerodicActivationToday { triggerCounters 3 }	Number of times a periodic trigger has been activated today.
triggerNumOfInterfaceActivationToday { triggerCounters 4 }	Number of times an interface trigger has been activated today.
triggerNumOfResourceActivationToday { triggerCounters 5 }	Number of times a CPU or memory trigger has been activated today.
triggerNumOfRebootActivationToday { triggerCounters 6 }	Number of times a reboot trigger has been activated today.
triggerNumOfPingPollActivationToday { triggerCounters 7 }	Number of times a ping-poll trigger has been activated today.
triggerNumOfStackMasterFailActivationToday { triggerCounters 8 }	Number of times a stack master fail trigger has been activated today.
triggerNumOfStackMemberActivationToday { triggerCounters 9 }	Number of times a stack member trigger has been activated today.

AT-LOOPPROTECT-MIB

The atLoopProtect-MIB (Figure 64-3, Table 64-16) defines objects for managing Loop Protection objects and triggers. Objects in this group have the object identifier atLoopProtect ({ modules 4 }).

Figure 64-3: The ATLoopProtect MIB Sub-tree

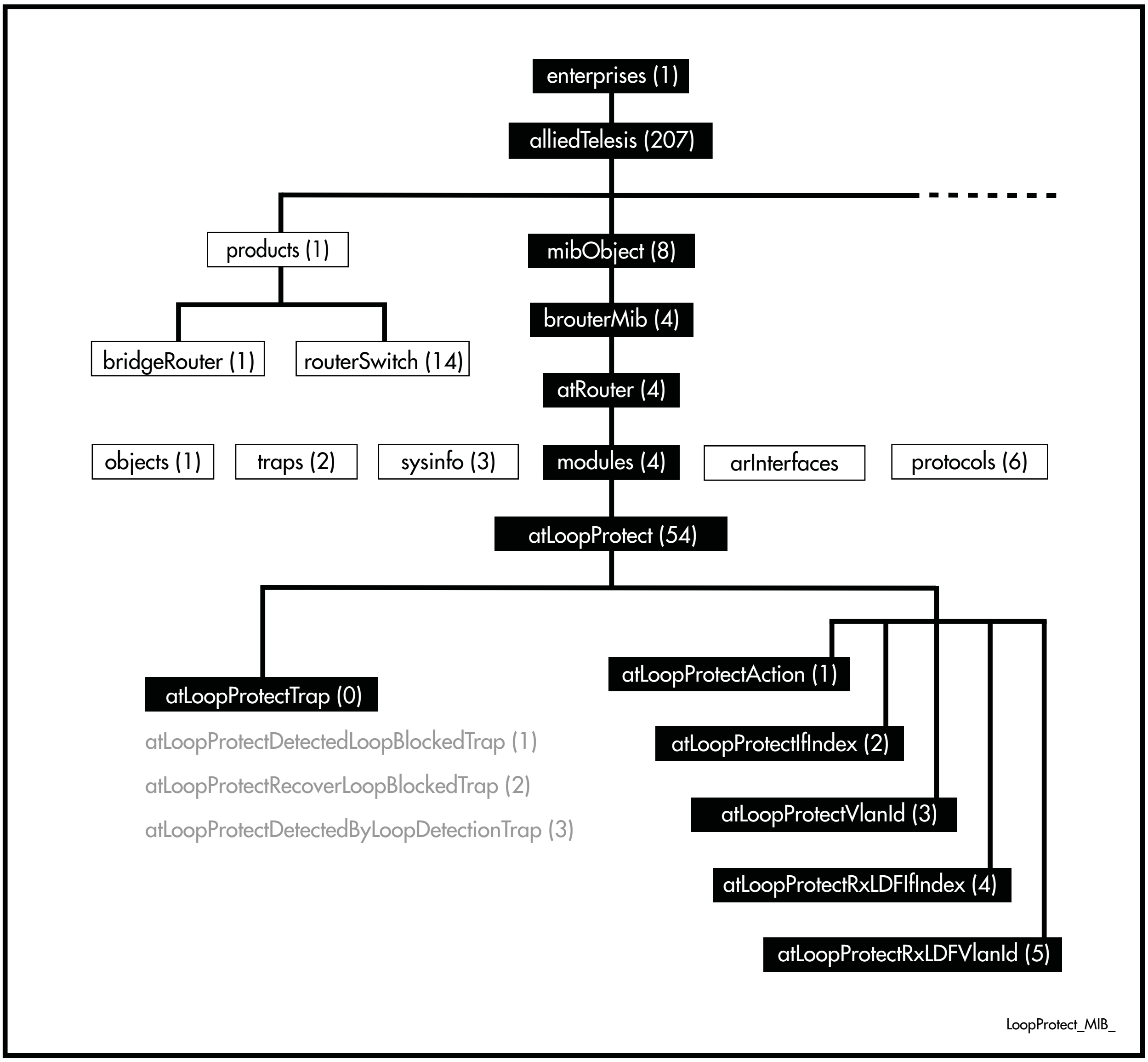


Table 64-16: Objects Defined in the AT-Loop Protect MIB

Object	**Object Identifier**	**Description**
{ atLoopProtect }	{ modules 54 }	The root of the Loop Protect object sub tree.
{ atLoopProtectTrap }	{ atLoopProtect0 }	The Loop Protection node state transition trap. List of traps (notifications) generated for Loop Protection.
{ atLoopProtectDetected LoopBlockedTrap }	{ atLoopProtectTrap1 }	Notification generated when the Loop Protection feature blocks an interface with a loop. The following bindings are associated with this trap: 1. atLoopProtectIfIndex 2. atLoopProtectVlanId 3. atLoopProtectAction
{ atLoopProtectRecover LoopBlockedTrap }	{ atLoopProtectTrap2 }	Notification generated when the Loop Protection feature restores a blocked interface back to normal operation. The following bindings are associated with this trap: 1. atLoopProtectIfIndex 2. atLoopProtectVlanId 3. atLoopProtectAction
{ atLoopProtectDetected ByLoopDetectionTrap }	{ atLoopProtectTrap3 }	Notification generated when the Loop Protection feature detects a loop by Loop Detection method. The following bindings are associated with this trap: 1. atLoopProtectIfIndex ■ atLoopProtectVlanId ■ atLoopProtectRxLDFIfIndex ■ atLoopProtectRxLDFVlanId
{ atLoopProtectAction }	{ atLoopProtect1 }	The Action for the Loop Protection feature. The following values are defined: 1. atLoopProtectAction-LearnDisable (0) 2. atLoopProtectAction-LearnEnable (1) 3. atLoopProtectAction-PortDisable (2) 4. atLoopProtectAction-PortEnable (3) 5. atLoopProtectAction-LinkDown (4) 6. atLoopProtectAction-LinkUp (5) 7. atLoopProtectAction-VlanDisable (6) 8. atLoopProtectAction-VlanEnable (7)
{ atLoopProtectIfIndex }	{ atLoopProtect2 }	The interface on which the loop was detected.
{ atLoopProtectVlanId }	{ atLoopProtect3 }	The VLAN ID on which the loop was detected.
{ atLoopProtectRxLDFIfIndex }	{ atLoopProtect4 }	The interface on which the loop detection frame was received.
{ atLoopProtectRxLDFVlanId }	{ atLoopProtect5 }	The VLAN ID on which the loop detection frame was received.

AT-SETUP-MIB

AT-SETUP-MIB defines objects for managing software installation and configuration files (Figure 64-4, Table 64-17). Objects in this group have the object identifier setup ({ modules 500 }). The procedure in Table 62-6 on page 62.22 shows how to use these MIB objects to upgrade to a new software version and boot configuration file. For objects used for file copying, see "AT-FILEv2-MIB" on page 64.62.

Figure 64-4: The AT-SETUP-MIB sub-tree

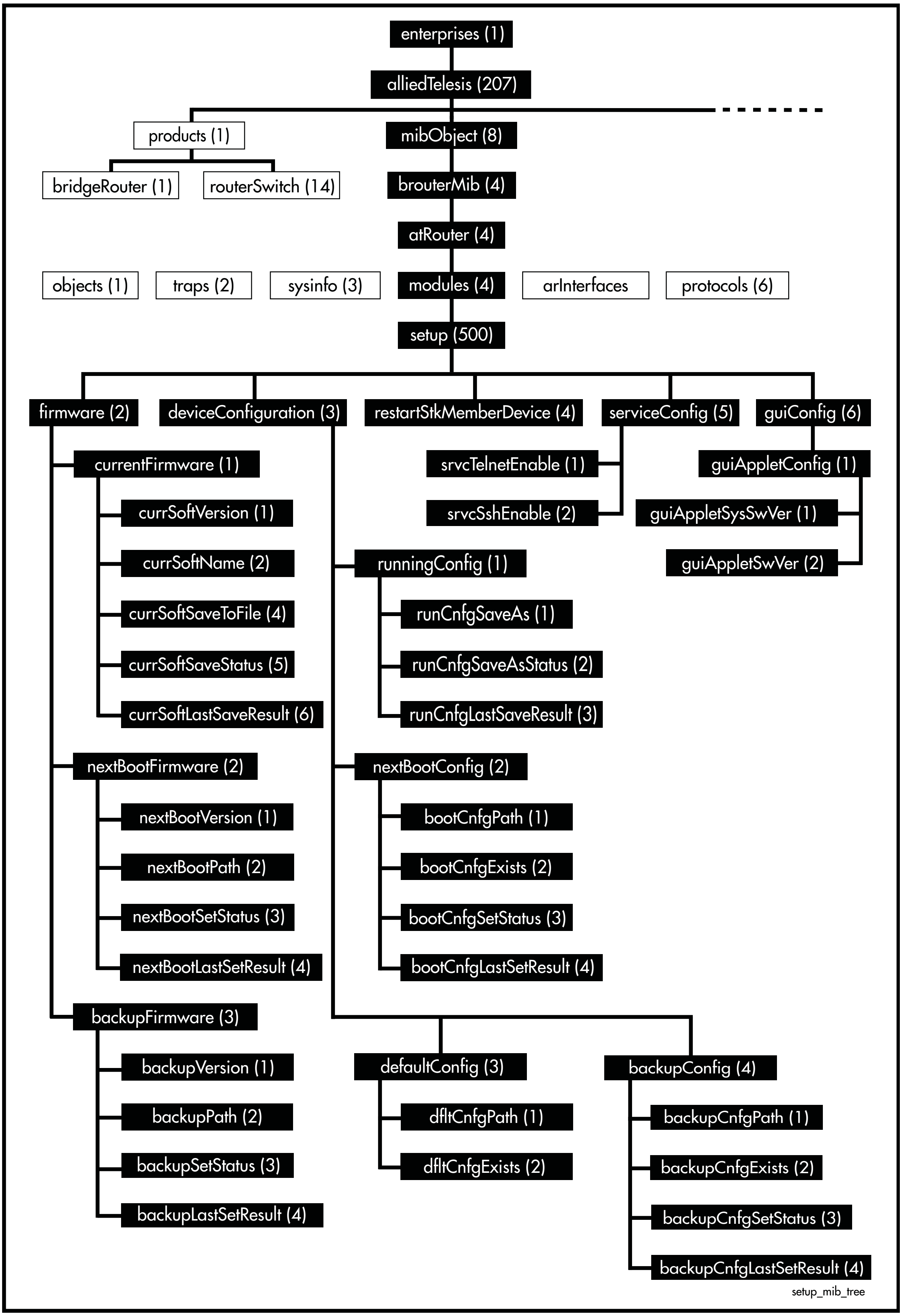


Table 64-17: Objects defined in AT-SETUP-MIB

Object	Object Identifier	Description
restartDevice	{ setup 1 }	Object for restarting the device. When set to '1', the device will restart immediately. **Note:** This object has been deprecated, use instead the restartStkMemberDevice object.
firmware	{ setup 2 }	Objects for managing the software version files that the device will install and run.
currentFirmware	{ firmware 1 }	Information about the current software version installed on the device.
currSoftVersion	{ currentFirmware 1 }	Current software version.
currSoftName	{ currentFirmware 2 }	Current software name.
currSoftSaveAs	{ currentFirmware 3 }	The file name to save the currently running software to the root of the Flash. Only one save operation can be executed at a time across all SNMP users. **Note:** This object has been deprecated, use instead the currSoftSaveToFile, currSoftSaveStatus and currSoftLastSaveResult objects.
currSoftSaveToFile	{ currentFirmware 4 }	Set with a URL to save the currently running software to the root of Flash or USB flash drive (e.g. 'flash:/filename.rel' or 'USB:/filename.rel'). The URL must not contain whitespace characters. Only one save operation can be executed at a time across all SNMP users and an operation may not be started unless the current value of currSoftSaveStatus is 'idle'. Immediately upon executing the set action, the actual firmware save operation is started and will continue on the device until it has completed or a failure occurs. When read, this object will return the URL of the last firmware save operation that was attempted.
currSoftSaveStatus	{ currentFirmware 5 }	This object will return the status of any current operation to store the running software to a release file. The following values may be returned: 1. (idle) - there is no release file save operation in progress 2. (success) - the last release file save operation completed successfully 3. (failure) - the last release file save operation failed 4. (saving) - a release file save operation is currently in progress When a read of this object returns a value of 'success' or 'failure', it will immediately be reset to 'idle' and a new operation may be initiated if desired. A detailed description of the last completed operation may be determined by reading currSoftLastSaveResult.
currSoftLastSaveResult	{ currentFirmware 6 }	Gives an indication of the result of the last completed SNMP operation to save the running firmware to a release file.

Table 64-17: Objects defined in AT-SETUP-MIB(cont.)

Object	Object Identifier	Description
nextBootFirmware	{ firmware 2 }	Information about the software version to be installed on the device when booting.
nextBootVersion	{ nextBootFirmware 1 }	Provides information on the software version (major.minor.interim, for example version 5.4.1) that the device will boot from. A zero will be returned if the version cannot be determined.
nextBootPath	{ nextBootFirmware 2 }	The full path to the release file that will be used the next time the device is rebooted. The URL must not contain whitespace characters. Only one set operation can be executed at a time across all SNMP users and an operation may not be started unless the current value of nextBootSetStatus is 'idle'. Immediately upon executing the set action, the system will attempt to set the new configuration path, and the process will continue on the device until it has completed or a failure occurs. This object can be set with an empty string in order to clear the current boot firmware. Otherwise, the path should be of the form ‘flash:/filename.cfg’ or 'card:/filename.cfg'. In order to set this object, the file must meet the following conditions: ■ it must exist ■ it must be located in the root of Flash (on the active master in a stacked environment) or USB flash drive ■ it must not be the same as the backup release file ■ it must have a .rel suffix ■ it must pass several internal checks to ensure that it is a genuine release file ■ in a stacked environment, there must be enough disk space available to store the release file on each stack member
nextBootSetStatus	{ nextBootFirmware 3 }	Returns the status of any current operation to set the next boot release file. The following values may be returned: ■ 1 (idle) - there is no boot release setting operation in progress ■ 2 (success) - the last boot release setting operation completed successfully ■ 3 (failure) - the last boot release setting operation failed ■ 5 (syncing) - a boot release setting operation is currently in progress and the file is being synchronized across the stack When a read of this object returns a value of 'success' or 'failure', it will immediately be reset to 'idle' and a new operation may be initiated if desired. A detailed description of the last completed operation may be determined by reading nextBootLastSetResult.
nextBootLastSetResult	{ nextBootFirmware 4 }	Gives an indication of the result of the last completed SNMP operation to set the boot release filename.

Table 64-17: Objects defined in AT-SETUP-MIB(cont.)

Object	Object Identifier	Description
backupFirmware	{ firmware 3 }	Information about the backup software version and path.
backupVersion	{ backupFirmware 1 }	Provides information on the backup software version (major.minor.interim, for example version 5.4.1) that the device will boot from. A zero will be returned if the version cannot be determined.
backupPath	{ backupFirmware 2 }	The full path to the backup release file that will be used the next time the device is rebooted. The URL must not contain whitespace characters. Only one set operation can be executed at a time across all SNMP users and an operation may not be started unless the current value of backupSetStatus is 'idle'. Immediately upon executing the set action, the system will attempt to set the new configuration path, and the process will continue on the device until it has completed or a failure occurs. This object can be set with an empty string in order to clear the current backup firmware. Otherwise, the path should be of the form 'flash:/filename.cfg' or 'card:/filename.cfg'. In order to set this object, the file must meet the following conditions: ■ it must exist ■ it must be located in the root of Flash (on the active master in a stacked environment) or USB flash drive ■ it must not be the same as the configured main release file ■ it must have a .rel suffix ■ it must pass several internal checks to ensure that it is a genuine release file ■ in a stacked environment, there must be enough disk space available to store the release file on each stack member
backupSetStatus	{ backupFirmware 3 }	Returns the status of any current operation to set the backup boot release file. The following values may be returned: ■ 1 (idle) - there is no backup boot release setting operation in progress ■ 2 (success) - the last backup boot release setting operation completed successfully ■ 3 (failure) - the last backup boot release setting operation failed ■ 5 (syncing) - a backup boot release setting operation is currently in progress and the file is being synchronized across the stack When a read of this object returns a value of 'success' or 'failure', it will immediately be reset to 'idle' and a new operation may be initiated if desired. A detailed description of the last completed operation may be determined by reading backupLastSetResult.
backupLastSetResult	{ backupFirmware 4 }	Gives an indication of the result of the last completed SNMP operation to set the backup boot release filename.
deviceConfiguration	{ setup 3	Objects for managing device configuration.
runningConfig	{ deviceConfiguration 1 }	

Table 64-17: Objects defined in AT-SETUP-MIB(cont.)

Object	Object Identifier	Description
runCnfgSaveAs	{ runningConfig 1 }	Set with a URL to save the currently running software to the root of Flash or USB flash drive (e.g. 'flash:/filename.rel' or 'usb:/filename.rel'). The URL must not contain whitespace characters. Only one set operation can be executed at a time across all SNMP users and an operation may not be started unless the current value of runCnfgSaveAsStatus is 'idle'. Immediately upon executing the set action, the system will attempt to save the running configuration and the process will continue on the device until it has completed or a failure occurs. When read, this object will return the URL of the last firmware save operation that was attempted.
runCnfgSaveAsStatus	{ runningConfig 2 }	Returns the status of any current operation to save the running configuration. The following values may be returned: 1. (idle) - there is no config file save operation in progress 2. (success) - the last config file save operation completed successfully 3. (failure) - the last config file save operation failed 4. (saving) - a config file save operation is currently in progress When a read of this object returns a value of 'success' or 'failure', it will immediately be reset to 'idle' and a new operation may be initiated if desired. A detailed description of the last completed operation may be determined by reading runCnfgLastSaveResult.
runCnfgLastSaveResult	{ runningConfig 3 }	Gives an indication of the result of the last completed SNMP operation to save the running configuration.
nextBootConfig	{ deviceConfiguration 2 }	
bootCnfgPath	{ nextBootConfig 1 }	The full path to the configuration file that will be used the next time the device is rebooted. The URL must not contain whitespace characters. Only one set operation can be executed at a time across all SNMP users and an operation may not be started unless the current value of bootCnfgSetStatus is 'idle'. Immediately upon executing the set action, the system will attempt to set the new configuration path, and the process will continue on the device until it has completed or a failure occurs. This object can be set with an empty string in order to clear the current boot configuration. Otherwise, the path should be of the form ‘flash:/myconfig.cfg’ or 'card:/filename.cfg'. In order to set this object, the file must meet the following conditions: ■ it must exist ■ it must be located in the root of Flash (on the active master in a stacked environment) or USB flash drive ■ it must have a .cfg suffix ■ in a stacked environment, there must be enough disk space available to store the configuration file on each stack member

Table 64-17: Objects defined in AT-SETUP-MIB(cont.)

Object	Object Identifier	Description
bootCnfgExists	{ nextBootConfig 2 }	This object will return the value TRUE if the currently defined boot configuration file exists, or FALSE if it does not.
bootCnfgSetStatus	{ nextBootConfig 3 }	Returns the status of any current operation to set the next boot configuration file. The following values may be returned: ■ 1 (idle) - there is no boot configuration setting operation in progress ■ 2 (success) - the last boot configuration setting operation completed successfully ■ 3 (failure) - the last boot configuration setting operation failed ■ 5 (syncing) - a boot configuration setting operation is currently in progress and the file is being synchronized across the stack When a read of this object returns a value of 'success' or 'failure', it will immediately be reset to 'idle' and a new operation may be initiated if desired. A detailed description of the last completed operation may be determined by reading bootCnfgLastSetResult.
bootCnfgLastSetResult	{ nextBootConfig 4 }	Gives an indication of the result of the last completed SNMP operation to set the boot configuration filename.
defaultConfig	{ deviceConfiguration 3 }	
dfltCnfgPath	{ defaultConfig 1 }	The full path of the configuration file to use as backup when the device is rebooted. This object is not settable. The default configuration file is always ‘flash:/default.cfg’.
dfltCnfgExists	{ defaultConfig 2 }	This object will return the value TRUE if the currently defined default configuration file exists, or FALSE if it does not.
backupConfig	{ deviceConfiguration 4 }	

Table 64-17: Objects defined in AT-SETUP-MIB(cont.)

Object	Object Identifier	Description
backupCnfgPath	{ backupConfig 1 }	The full path to the backup configuration file that will be used the next time the device is rebooted. The URL must not contain whitespace characters. Only one set operation can be executed at a time across all SNMP users and an operation may not be started unless the current value of backupCnfgSetStatus is 'idle'. Immediately upon executing the set action, the system will attempt to set the new backup configuration path, and the process will continue on the device until it has completed or a failure occurs. This object can be set with an empty string in order to clear the current boot configuration. Otherwise, the path should be of the form 'flash:/myconfig.cfg' or 'card:/ filename.cfg'. In order to set this object, the file must meet the following conditions: ■ it must exist ■ it must be located in the root of Flash (on the active master in a stacked environment) or USB flash drive ■ it must have a .cfg suffix ■ in a stacked environment, there must be enough disk space available to store the configuration file on each stack member
backupCnfgExists	{ backupConfig 2 }	This object will return the value TRUE if the currently defined backup configuration file exists, or FALSE if it does not.
backupCnfgSetStatus	{ backupConfig 3 }	Returns the status of any current operation to set the next backup boot configuration file. The following values may be returned: ■ 1 (idle) - there is no backup boot configuration setting operation in progress ■ 2 (success) - the last backup boot configuration setting operation completed successfully ■ 3 (failure) - the last backup boot configuration setting operation failed ■ 5 (syncing) - a backup boot configuration setting operation is currently in progress and the file is being synchronized across the stack When a read of this object returns a value of 'success' or 'failure', it will immediately be reset to 'idle' and a new operation may be initiated if desired. A detailed description of the last completed operation may be determined by reading backupCnfgLastSetResult.
backupCnfgLastSetResult	{ backupConfig 4 }	Gives an indication of the result of the last completed SNMP operation to set the backup boot configuration filename.

Table 64-17: Objects defined in AT-SETUP-MIB(cont.)

Object	Object Identifier	Description
restartStkMemberDevice	{ setup 4 }	This object causes a specified device to restart immediately. The restart is initiated by setting its value to the device's stack member ID. Setting its value to zero will cause all devices in the stack, or a standalone device, to restart. Reading the object will always return zero.
serviceConfig	{ setup 5 }	
srvcTelnetEnable	{ serviceConfig 1 }	This object is used to either read or set the state of the telnet server on a device. Telnet can be enabled by setting the value of this object to ‘enable(1)’ or can be disabled by setting the value ‘disable(2)’.
srvcSshEnable	{ serviceConfig 2 }	This object is used to either read or set the state of the SSH server on a device. SSH can be enabled by setting the value of this object to ‘enable(1)’ or can be disabled by setting the value ‘disable(2)’.
guiConfig	{ setup 6 }	
guiAppletConfig	{ guiConfig 1 }	
guiAppletSysSwVer	{ guiAppletConfig 1 }	This object represents the system software release that the currently selected GUI applet was designed to run on. The system automatically searches for GUI applet files that reside in the root directory of the Flash memory, and selects the latest available file that is applicable to the currently running system software. This is the applet that will be uploaded to a user's web browser when they initiate the GUI.
	{ guiAppletConfig 2 }	This object represents the software version of the currently selected GUI applet. The system automatically searches for GUI applet files residing in the root directory of the Flash memory, and selects the latest available one that is applicable to the currently running system software. This is the applet that will be uploaded to a user's web browser when they initiate the GUI.

AT-DNS-CLIENT-MIB

AT-DNS-CLIENT-MIB contains definitions of managed objects for the Allied Telesis DNS Client Configuration.

Objects in this group have the object identifier atDns ({ Modules 501 }). Table 64-18 lists the objects supported by the AlliedWare Plus^{TM} Operating System.

Table 64-18: Objects defined in AT-DNS-CLIENT-MIB

Object	Object Identifier	Description
atDnsClient	{ atDns 1 }	MIB File for DNS Client Configuration.
atDNSServerIndexNext	{ atDnsClient 1 }	The next available value for the object 'atDNSServerIndex'. The value is used by a management application to create an entry in the 'atDNSServerTable'.
atDNSServerTable	{ atDnsClient 2 }	Table of information about the Domain Name System (DNS) Server configurations in the system, indexed by 'atDNSServerIndex'.
atDNSServerEntry	{ atDNSServerTable 1 }	Information about a single DNS Server Configuration.
atDNSServerIndex	{ atDNSServerEntry 1 }	The index corresponding to the particular DNS Server Configuration. When creating a new entry in the table, the value of this object must be equal to the value in the 'atDNSServerIndexNext'.
atDNSServerAddrType	{ atDNSServerEntry 2 }	The Internet Address Type of the 'atDNSServerAddr' object. Can be one of the following: ■ unknown (0) ■ ipv4 (1) - default ■ ipv6 (2) - not supported ■ ipv4z (3) - not supported ■ ipv6z (4) - not supported ■ dns (16) - not supported
atDNSServerAddr	{ atDNSServerEntry 3 }	The IP Address of the DNS Server. When a new entry is created, this object is set to the default of '0.0.0.0' { '00000000'h }. The management application will change this to the desired value using a SET operation.
atDNSServerStatus	{ atDNSServerEntry 4 }	The status of the current entry (row). Can be one of the following: ■ active (1) ■ createAndGo (4) ■ destroy (6) To create a new entry the management application must set this object with value 'createAndGo (4)'. To delete an entry, the management application must set this object with value 'destroy (6)'. Once an entry is deleted, all subsequent entries in the table will be renumbered. The default is 1 (active)

AT-NTP-MIB

This MIB contains objects for managing the Allied Telesis Network Time Protocol (NTP) configuration (Table 64-19). The objects reside in the module atNtp { modules 502 }, organized in the following groups:

- NTP Peer/Server Table - a table containing information on the Network Time Protocol (NTP) peers or server configurations in the system.
- Associations Table - a list of installed software; used also to remove software from the device.
- Status Table - Objects in this group are not supported.

Table 64-19: Objects defined in AT-NTP-MIB

Object	Object Identifier	Description
atNtp	{ modules 502 }	MIB containing objects for configuring NTP.
atNtpPeerIndexNext	{ atNtp 6 }	The next available index number to be used for object 'atNtpPeerIndex'.
atNtpPeerTable	{ atNtp 7 }	Table containing information on the Network Time Protocol (NTP) peers or server configurations in the system. Indexed by: ■ atNtpPeerIndex
atNtpPeerEntry	{ atNtpPeerTable 1 }	Information about a single NTP server or peer configuration.
atNtpPeerIndex	{ atNtpPeerEntry 1 }	The index number corresponding to a particular NTP server or peer configuration in the system. To create a new entry, the value of this object should be the same as that of the value of atNtpPeerIndexNext object, otherwise the entry creation will fail.
atNtpPeerNameAddr	{ atNtpPeerEntry 2 }	The host name, or the IP address of the NTP peer. When a new row (entry) is created, this object is set with a default of '0.0.0.0', and the management application should change it to a desired value by using a SET operation.
atNtpPeerMode	{ atNtpPeerEntry 3 }	The mode of the peer. Can be one of the following: ■ server (1) ■ peer (2) - default
atNtpPeerPreference	{ atNtpPeerEntry 4 }	The values in this object specifies whether this peer is the preferred one. Valid values are 0 to 2: ■ 0 - unknown - default ■ 1 - not preferred ■ 2 - preferred When the value is 'not preferred' (1) NTP chooses the peer with which to synchronize the time on the local system. If the object is set to 'preferred' (2) NTP will choose the corresponding peer to synchronize the time with.

Table 64-19: Objects defined in AT-NTP-MIB(cont.)

Object	Object Identifier	Description
atNtpPeerVersion	{ atNtpPeerEntry 5 }	The NTP version the peer supports. Can be one of the following: ■ 0 - unknown - default ■ 1 - version 1 ■ 2 - version 2 ■ 3 - version 3 ■ 4 - version 4
atNtpPeerKeyNumber	{ atNtpPeerEntry 6 }	The authentication key number. Default number is 0.
atNtpPeerRow Status	{ atNtpPeerEntry 7 }	The current status of this peer entry. The following values may be returned when reading this object: ■ active (1) - this value is returned on reading of this entry. ■ createAndGo (4) - this value is set by the management application when creating a new entry ■ destroy (6) - value set by the management application when deleting the entry. When an entry is deleted, all subsequent entries in the table will be re-indexed.
atNtpAssociationTable	{ atNtp 10 }	Table containing information on the Network Time Protocol (NTP) associations. Indexed by: ■ atNtpAssociationIndex
atNtpAssociationEntry	{ atNtpAssociationTable 1 }	Information about a single NTP server or peer configuration.
atNtpAssociationIndex	{ atNtpAssociationEntry 1 }	The index number corresponding to a particular NTP server or peer configuration in the system. To create a new entry, the value of this object should be the same as that of the value of atNtpPeerIndexNext object, otherwise the entry creation will fail.
atNtpAssociationPeerAddr	{ atNtpAssociationEntry 2 }	The host name, or the IP address of the NTP peer. When a new row (entry) is created, this object is set with a default of '0.0.0.0', and the management application should change it to a desired value by using a SET operation.
atNtpAssociationStatus	{ atNtpAssociationEntry 3 }	The status of this association. Can be one of the following: ■ master (synced) ■ master (unsynced) ■ selected ■ candidate ■ configured ■ unknown
atNtpAssociationConfigured	{ atNtpAssociationEntry 4 }	The value in this object specifies whether the association is from configuration or not. Value can be: ■ configured ■ dynamic
atNtpAssociationRefClkAddr	{ atNtpAssociationEntry 5 }	The IP Address for the reference clock.
atNtpAssociationStratum	{ atNtpAssociationEntry 6 }	The stratum of the peer clock.

Table 64-19: Objects defined in AT-NTP-MIB(cont.)

Object	Object Identifier	Description
atNtpAssociationPoll	{ atNtpAssociationEntry 7 }	The time between NTP requests from the device to the server, in seconds.
atNtpAssociationReach	{ atNtpAssociationEntry 8 }	An integer that indicates the reachability status of the peer.
atNtpAssociationDelay	{ atNtpAssociationEntry 9 }	The round trip delay between the device and the server.
atNtpAssociationOffset	{ atNtpAssociationEntry 10 }	The difference between the device clock and the server clock.
atNtpAssociationDisp	{ atNtpAssociationEntry 11 }	The lowest measure of error associated with peer offset, based on delay, in seconds.
atNtpStatus	{ atNtp 11 }	Group of objects containing system status information. The objects in this group are not supported.
atNtpSysClockSync	{ atNtpStatus 1 }	Not supported.
atNtpSysStratum	{ atNtpStatus 2 }	Not supported.
atNtpSysReference	{ atNtpStatus 3 }	Not supported.
atNtpSysFrequency	{ atNtpStatus 4 }	Not supported.
atNtpSysPrecision	{ atNtpStatus 5 }	Not supported.
atNtpSysRefTime	{ atNtpStatus 6 }	Not supported.
atNtpSysClkOffset	{ atNtpStatus 7 }	Not supported.
atNtpSysRootDelay	{ atNtpStatus 8 }	Not supported.
atNtpSysRootDisp	{ atNtpStatus 9 }	Not supported.

AT-EPSRv2-MIB

The EPSRv2 Group-MIB defines objects for managing Epsrv2 objects and triggers (Figure 64-5, Table 64-20). Objects in this group have the object identifier Epsrv2 ({ modules 536 }).

Figure 64-5: The AT-EPSRv2 MIB sub-tree

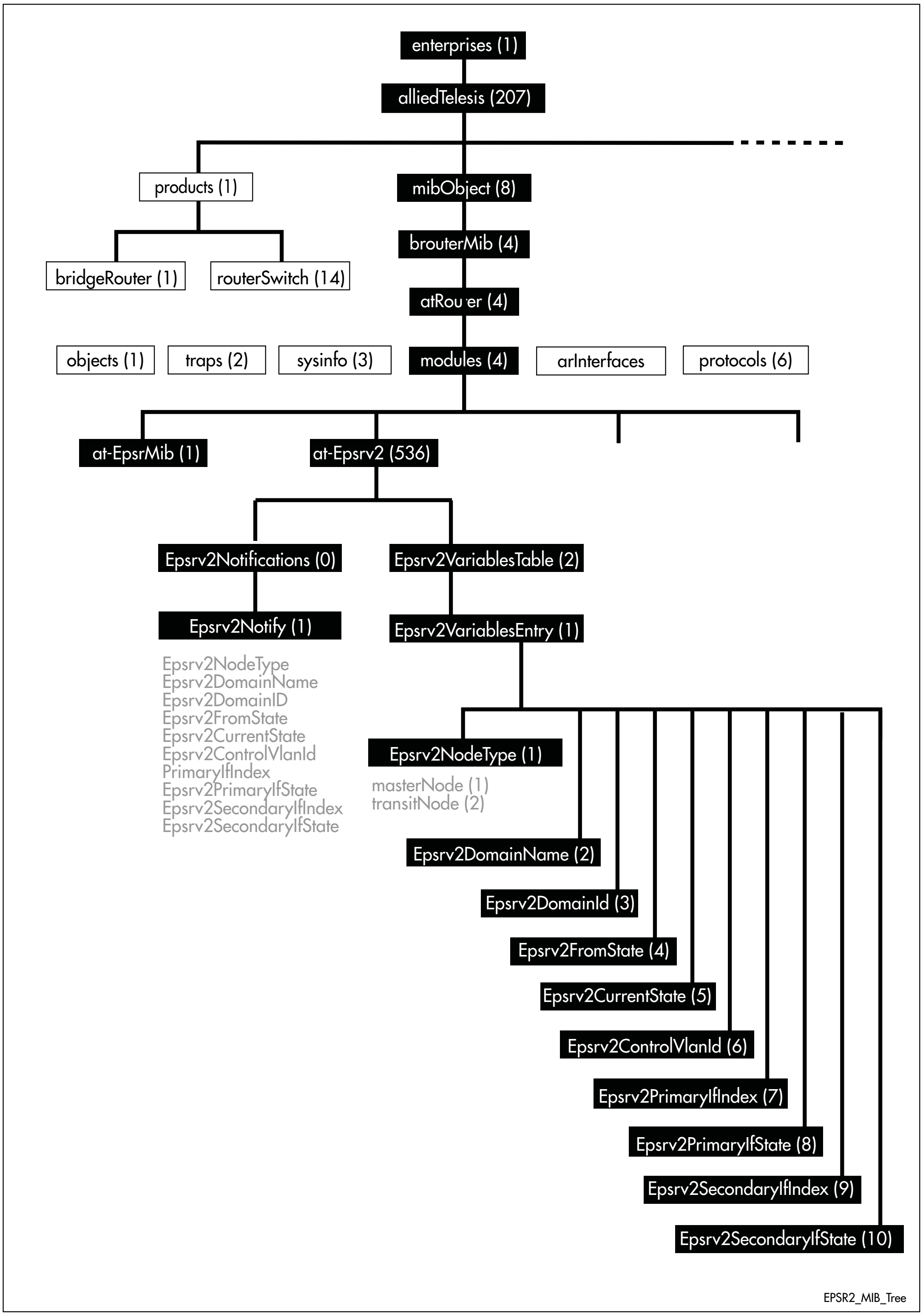


Table 64-20: atEpsrv2Objects Defined in the AT-EPSRV2 MIB

Object	Object Identifier	Description
{ at-Epsrv2 }	{ modules 536 }	The root of the Epsrv2 object sub tree.
{ atEpsrv2Notifications }	{ at-Epsrv2 0 }	
{ atEpsrv2Notify }	{ atEpsrv2Notifications 1 }	EPSR Master/Transit node state transition trap. Note that there is a one to one relationship between nodes and domains.
{ Epsrv2NodeType }	{ atEpsrv2VariablesEntry 1 }	The EPSR node type: either master or transit.
{ atEpsrv2DomainName }	{ atEpsrv2VariablesEntry 2 }	The name of the EPSR domain.
{ atEpsrv2DomainID }	{ atEpsrv2VariablesEntry 3 }	The ID of the EPSR domain.
{ Epsrv2FromState }	{ atEpsrv2VariablesEntry 4 }	The previous state of the EPSR domain
{ Epsrv2Current State }	{ atEpsrv2VariablesEntry 5 }	The current state of the EPSR domain.
{ Epsrv2ControlVlanId }	{ atEpsrv2VariablesEntry 6 }	The VLAN identifier for the control VLAN.
{ Epsrv2PrimaryIfIndex }	{ atEpsrv2VariablesEntry 7 }	The IfIndex of the primary interface.
{ atEpsrv2PrimaryIfState }	{ atEpsrv2VariablesEntry 8 }	The current state of the primary interface.
{ atEpsrv2SecondaryIfIndex }	{ atEpsrv2VariablesEntry 9 }	The IfIndex of the secondary interface.
{ atEpsrv2SecondaryIfState }	{ atEpsrv2VariablesEntry 10 }	The state of the secondary interface.
{ atEpsrv2VariablesTable }	{ at-Epsrv2 2 }	The enterprise Epsrv2VariablesTable.
{ atEpsrv2VariablesEntry }	{ atEpsrv2VariablesTable 1}	Contains entries within the enterprise atEpsrv2VariablesTable.
{ atEpsrv2NodeType }	{ atEpsrv2VariablesEntry 1 }	The EPSR domain node type: either 1. master (1) 2. transit (2)
{ atEpsrv2DomainName }	{ Epsrv2NodeType 2 }	The name of the EPSR domain.
{ atEpsrv2DomainID }	{ Epsrv2NodeType 3 }	The ID of the EPSR domain.
{ atEpsrv2FromState }	{ Epsrv2NodeType 4 }	The previous state of the EPSR domain
{ atEpsrv2Current State }	{ Epsrv2NodeType 5 }	The current state of the EPSR domain.
{ atEpsrv2ControlVlanId }	{ Epsrv2NodeType 6 }	The VLAN identifier for the control VLAN.
{ Epsrv2PrimaryIfIndex }	{ Epsrv2NodeType 7 }	The IfIndex of the primary interface.
{ atEpsrv2PrimaryIfState }	{ Epsrv2NodeType 8 }	The current state of the primary interface.
{ atEpsrv2SecondaryIfIndex }	{ Epsrv2NodeType 9 }	The IfIndex of the secondary interface.
{ atEpsrv2SecondaryIfState }	{ Epsrv2NodeType 10 }	The state of the secondary interface.
TEXTUAL CONVENTIONS		
{ atEpsrv2NodeState }		The trap states that can be advertised for an EPSR domain node. The following states are defined: 1. idle (1) 2. complete (2) 3. failed (3) 4. linksUp (4) 5. linksDown (5) 6. preForward (6) 7. unknown (7)

Table 64-20: atEpsrv2Objects Defined in the AT-EPSRV2 MIB(cont.)

Object	Object Identifier	Description
{ atEpsrv2InterfaceState }		The trap states that can be advertised for an EPSR interface. The following states are defined: 1. unknown (1) 2. down (2) 3. blocked (3) 4. forward (4)

AT-DHCPSN-MIB

This MIB contains objects for displaying and managing DHCP snooping and ARP security information on the switch. (Table 64-21). The objects reside in the module atDhcpsn { modules 537 }, organized in the following groups:

- The DHCP Snooping Events group (atDhcpsnEvents) contains notifications (traps)
- The DHCP Snooping table (atDhcpsnVariablesTable) contains DHCP snooping information
- The ARP Security table (atArpsecVariablesTable) contains ARP security information

Table 64-21: Objects defined in AT-DHCPSN-MIB

Object	Object Identifier	Description
atDhcpsn	{ modules 537 }	This MIB file contains definitions of managed objects for DHCP Snooping in AlliedWare Plus^{TM}.
atDhcpsnEvents	{ atDhcpsn 1 }	DHCP Snooping notifications (traps)
atDhcpsnTrap	{ atDhcpsnEvents 1 }	DHCP Snooping violation notification.
atArpsecTrap	{ atDhcpsnEvents 2 }	DHCP Snooping ARP Security violation notification.
atDhcpsnVariablesTable	{ atDhcpsn 1 }	The DHCP Snooping table. This table contains rows of DHCP Snooping information.
atDhcpsnVariablesEntry	{ atDhcpsnVariablesTable 1 }	A set of parameters that describe the DHCP Snooping features.
atDhcpsnIfIndex	{ atDhcpsnVariablesEntry 1 }	Ifindex of the port that the packet was received on.
atDhcpsnVid	{ atDhcpsnVariablesEntry 2 }	VLAN ID of the port that the packet was received on.
atDhcpsnSmac	{ atDhcpsnVariablesEntry 3 }	Source MAC address of the packet that caused the trap.
atDhcpsnOpcode	{ atDhcpsnVariablesEntry 4 }	Opcode value of the BOOTP packet that caused the trap. Only bootpRequest(1) or bootpReply(2) is valid.
atDhcpsnCiaddr	{ atDhcpsnVariablesEntry 5 }	Ciaddr value of the BOOTP packet that caused the trap.
atDhcpsnYiaddr	{ atDhcpsnVariablesEntry 6 }	Yiaddr value of the BOOTP packet that caused the trap.
atDhcpsnGiaddr	{ atDhcpsnVariablesEntry 7 }	Giaddr value of the BOOTP packet that caused the trap.
atDhcpsnSiaddr	{ atDhcpsnVariablesEntry 8 }	Siaddr value of the BOOTP packet that caused the trap.
atDhcpsnChaddr	{ atDhcpsnVariablesEntry 9 }	Chaddr value of the BOOTP packet that caused the trap.

Table 64-21: Objects defined in AT-DHCPSN-MIB(cont.)

Object	Object Identifier	Description
atDhcpsnVioType	{ atDhcpsnVariablesEntry 10 }	The reason that the trap was generated. ■ invalidBootp(1) indicates that the received BOOTP packet was invalid. For example, it is neither BootpRequest nor BootpReply. ■ invalidDhcpAck(2) indicates that the received DHCP ACK was invalid. ■ invalidDhcpRelDec(3) indicates the DHCP Release or Decline was invalid. ■ invalidIp(4) indicates that the received IP packet was invalid. ■ maxBindExceeded(5) indicates that if the entry was added, the maximum bindings configured for the port would be exceeded. ■ opt82InsertErr(6) indicates that the insertion of Option 82 failed. ■ opt82RxInvalid(7) indicates that the received Option 82 information was invalid. ■ opt82RxUntrusted(8) indicates that Option 82 information was received on an untrusted port. ■ opt82TxUntrusted(9) indicates that Option 82 would have been transmitted out an untrusted port. ■ replyRxUntrusted(10) indicates that a BOOTP Reply was received on an untrusted port. ■ srcMacChaddrMismatch(11) indicates that the source MAC address of the packet did not match the BOOTP CHADDR of the packet. ■ staticEntryExisted(12) indicates that the static entry to be added already exists. ■ dbAddErr(13) indicates that adding an entry to the database failed.
atArpsecVariablesTable	{ atDhcpsn 2 }	The ARP Security table. This table contains rows of DHCP Snooping ARP Security information.
atArpsecVariablesEntry	{ atArpsecVariablesTable 1 }	A set of parameters that describe the DHCP Snooping ARP Security features.
atArpsecIfIndex	{ atArpsecVariablesEntry 1 }	Ifindex of the port that the ARP packet was received on.
atArpsecClientIP	{ atArpsecVariablesEntry 2 }	Source IP address of the ARP packet.
atArpsecSrcMac	{ atArpsecVariablesEntry 3 }	Source MAC address of the ARP packet.
atArpsecVid	{ atArpsecVariablesEntry 4 }	VLAN ID of the port that the ARP packet was received on.

Table 64-21: Objects defined in AT-DHCPSN-MIB(cont.)

Object	Object Identifier	Description
atArpsecVioType	{ atArpsecVariablesEntry 5 }	The reason that the trap was generated. ■ srcIpNotFound(1) indicates that the Sender IP address of the ARP packet was not found in the DHCP Snooping database. ■ badVLAN(2) indicates that the VLAN of the DHCP Snooping binding entry associated with the Sender IP address of the ARP packet does not match the VLAN that the ARP packet was received on. ■ badPort(3) indicates that the port of the DHCP Snooping binding entry associated with the Sender IP address of the ARP packet does not match the port that the ARP packet was received on. ■ srcIpNotAllocated(4) indicates that the CHADDR of the DHCP Snooping binding entry associated with the Sender IP address of the ARP packet does not match the Source MAC and/or the ARP source MAC of the ARP packet.

AT-FILEv2-MIB

This MIB contains objects for displaying and managing file content of Flash, USB storage devices and NVS, and copying, moving and deleting files from local and remote sources (Table 64-22).

The objects reside in the module atFilev2 { modules 600 }, organized in the following groups:

- The file operation devices - object for various devices supported for file operations
- The File Info Table - information about all files, including pathnames, that are present on the device
- The USB storage device table - information about the USB storage device configured on the device

The procedure in "Copy a File to or from a TFTP Server" on page 62.20 shows how to use these MIB objects to upgrade to a new software version and boot configuration file.

Table 64-22: Objects defined in AT-FILEv2-MIB

Object	Object Identifier	Description
atFilev2	{ modules 600 }	MIB containing objects for listing and managing files.
atFilev2FileOperation	{ atFilev2 3 }	Collection of file operation objects available for configuration, to enable copying, moving and deleting files.
atFilev2SourceStackID	{ atFilev2Operation 1 }	Specifies the Stack ID of the source file. Set an integer corresponding to the stack ID of the stack member to use as the source. For devices that are not capable of being stacked, set with the value 1. This value is ignored if the source device is set to TFTP.
atFilev2SourceDevice	{ atFilev2Operation 2 }	Specifies the source device for the file to be copied. Valid values are 1 to 4. Set a value that corresponds with the various devices, as below: ■ 1 - Flash - default ■ 2 - Card - not supported ■ 3 - NVS ■ 4 - TFTP ■ 5 - USB For copying files, you may use any combination of devices for the source and destination, except for copying from TFTP to TFTP. For moving files you cannot use TFTP as source or destination. For deleting files, the source cannot be TFTP. You must fully configure all required parameters before an operation can commence. Where a TFTP operation is configured, an IP address must also be set via atFilev2TftpIPAddr. To copy a file from TFTP to Flash, use 4 for source and 1 for destination.
atFilev2SourceFilename	{ atFilev2Operation 3 }	Specifies the filename of the source file to copy, move or delete. Include any path as required, but the storage type is not necessary. For example, to copy the file `latest.cfg` from the backupconfigs/routers directory on the TFTP server, you would set: `backupconfigs/routers/latest.cfg`

Table 64-22: Objects defined in AT-FILEv2-MIB(cont.)

Object(cont.)	**Object Identifier**	**Description**
atFilev2DestinationStackID	{ atFilev2Operation 4 }	Specifies the Stack ID for the destination file. For devices that are not capable of being stacked, set with the value 1. This value is ignored if the destination device is set to TFTP, or if a deletion operation is carried out.
atFilev2DestinationDevice	{ atFilev2Operation 5 }	Specifies the destination device for the files to be copied into. Valid values are 1 to 4. Set a value that corresponds with the various devices, as below: ■ 1 - Flash - default ■ 2 - Card - not supported ■ 3 - NVS ■ 4 - TFTP ■ 5 - USB For copying files, you may use any combination of devices for the source and destination, except for copying from TFTP to TFTP. For moving files you cannot use TFTP as source or destination. For deleting files, this object is ignored. You must fully configure all required parameters before an operation can commence. Where a TFTP operation is configured, an IP address must also be set via atFilev2TftpIPAddr. To copy a file from TFTP to Flash, use 4 for source and 1 for destination.
atFilev2DestinationFilename	{ atFilev2Operation 6 }	Specifies the destination filename of the file to be copied or moved. Include any path as required, but the storage type is not necessary. The destination filename does not need to be the same as the source filename, and this object is ignored for file deletion operations. For example, to copy a release file from the TFTP server to the backup release directory on Flash, you would set: `backuprelease/latest.rel` Note: If the destination is set to Flash, card or NVS, any file at the destination that shares the destination filename will be overwritten by a move or copy operation.

Table 64-22: Objects defined in AT-FILEv2-MIB(cont.)

Object(cont.)	Object Identifier	Description
atFilev2CopyBegin	{ atFilev2Operation 7 }	Represents the status of the copy file operation, in the form of octet string. A read on this object can return several possible values, depending on the current status of the system and the various file operation objects: ■ idle - There is no file operation in progress and all required objects have been set correctly. Setting a '1' to this object will begin the file copy. ■ Error codes: [1-7] - A copy operation cannot be started until these errors are resolved. See below for key. ■ [action]ing x [--> y] - A file operation is currently in progress. You cannot start another operation while the object is returning this value. ■ [action] x [--> y] success - The last copy, move or delete operation was successfully completed. ■ [action] x [--> y] failure: [err] - The last copy, move or delete operation failed, with the error message attached. Common failures include lack of space on the destination file system, incorrect source file names or communication errors with remote services. Upon reading a success or failure message, the message will be cleared and the next read will result in either an 'idle' message or an 'Error codes' message if not all required objects have been correctly set. If the read returned 'idle', a new file operation can now be started. Following are possible values returned as Error codes for file copy: ■ 1 - atFilev2SourceDevice has not been set ■ 2 - atFilev2SourceFilename has not been set ■ 3 - atFilev2DestinationDevice has not been set ■ 4 - atFilev2DestinationFilename has not been set ■ 5 - atFilev2SourceDevice and atFilev2DestinationDevice are both set to TFTP ■ 6 - the combination of source device, stackID and filename is the same as the destination device, stackID and filename (i.e. it is not valid to copy a file onto itself. ■ 7 - TFTP IP address has not been set and TFTP has been set for one of the devices Provided all above requirements are met, immediately upon executing the SNMP set, the device will indicate that it was a success. The actual file copy itself will be started and continue on the device until it has completed. For large files, operations can take several minutes to complete. Subsequent reads of the object will return one of messages shown in the first table, to allow for tracking of the progress of the copy operation.

Table 64-22: Objects defined in AT-FILEv2-MIB(cont.)

Object(cont.)	Object Identifier	Description
atFilev2MoveBegin	{ atFilev2Operation 8 }	Represents the status of the move file operation, in the form of octet string. A read on this object can return several possible values, depending on the current status of the system and the various file operation objects: ■ idle - There is no file operation in progress and all required objects have been set correctly. Setting a '1' to this object will begin the file move. ■ Error codes: [1-6] - A move operation cannot be started until these errors are resolved. See below for key. ■ [action]ing x [--> y] - A file operation is currently in progress. You cannot start another operation while the object is returning this value. ■ [action] x [--> y] success - The last copy, move or delete operation was successfully completed. ■ [action] x [--> y] failure: [err] - The last copy, move or delete operation failed, with the error message attached. Common failures include lack of space on the destination file system, incorrect source file names or communication errors with remote services. Upon reading a success or failure message, the message will be cleared and the next read will result in either an 'idle' message or an 'Error codes' message if not all required objects have been correctly set. If the read returned 'idle', a new file operation can now be started. Following are possible values returned as Error codes for file move: ■ 1 - atFilev2SourceDevice has not been set ■ 2 - atFilev2SourceFilename has not been set ■ 3 - atFilev2DestinationDevice has not been set ■ 4 - atFilev2DestinationFilename has not been set ■ 5 - either atFilev2SourceDevice or atFilev2DestinationDevice are set to TFTP ■ 6 - the combination of source device, stackID and filename is the same as the destination device, stackID and filename (i.e. it is not valid to move a file onto itself. Provided all above requirements are met, immediately upon executing the SNMP set, the device will indicate that it was a success. The actual file move itself will be started and continue on the device until it has completed. For large files, operations can take several minutes to complete. Subsequent reads of the object will return one of messages shown in the first table, to allow for tracking of the progress of the move operation.

Table 64-22: Objects defined in AT-FILEv2-MIB(cont.)

Object(cont.)	Object Identifier	Description
atFilev2DeleteBegin	{ atFilev2Operation 9 }	Represents the status of the delete file operation, in the form of octet string. A read on this object can return several possible values, depending on the current status of the system and the various file operation objects: ■ idle - There is no file operation in progress and all required objects have been set correctly. Setting a '1' to this object will begin the file deletion. ■ Error codes: [1-3] - A delete operation cannot be started until these errors are resolved. See below for key. ■ [action]ing x [--> y] - A file operation is currently in progress. You cannot start another operation while the object is returning this value. ■ [action] x [--> y] success - The last copy, move or delete operation was successfully completed. ■ [action] x [--> y] failure: [err] - The last copy, move or delete operation failed, with the error message attached. Common failures include lack of space on the destination file system, incorrect source file names or communication errors with remote services. Upon reading a success or failure message, the message will be cleared and the next read will result in either an 'idle' message or an 'Error codes' message if not all required objects have been correctly set. If the read returned 'idle', a new file operation can be started. File deletion operations ignore the values set in the atFilev2DestinationStackID, atFilev2DestinationDevice and atFilev2DestinationFilename objects. The file deletion operation is equivalent to the CLI 'delete force [file]' command, so it is possible to delete any normally-protected system files, such as the currently configured boot release. Following are possible values returned as Error codes for file move: ■ 1 - atFilev2SourceDevice has not been set ■ 2 - atFilev2SourceFilename has not been set ■ 3 - atFilev2SourceDevicehas not been set to TFTP Provided all above requirements are met, immediately upon executing the SNMP set, the device will indicate that it was a success. The actual file move itself will be started and continue on the device until it has completed. For large files, operations can take several minutes to complete. Subsequent reads of the object will return one of messages shown in the first table, to allow for tracking of the progress of the move operation.
atFilev2Flash_1	{ atFilev2Operation 10 }	Represents the Flash operation device object
atFilev2Card_2	{ atFilev2Operation 11 }	Represents the Card operation device object
atFilev2Nvs_3	{ atFilev2Operation 12 }	Represents the NVS operation device object

Table 64-22: Objects defined in AT-FILEv2-MIB(cont.)

Object(cont.)	Object Identifier	Description
atFilev2Tftp_4	{ atFilev2Operation 13 }	Represents the TFTP operation device object
atFilev2TftpIPAddr	{ atFilev2Tftp_4 1 }	The IP address of the TFTP server that is to be used for the file copy process. This IP Address needs to be reachable from the device, or the file copy will fail.
atFilev2Usb	{ atFilev2Operation 15 }	Represents the USB storage device operation device object.
atFilev2InfoTable	{ atFilev2 5 }	A list of all files, including pathnames, that are present on the device. Hidden system files are not displayed.
atFilev2InfoEntry	{ atFilev2InfoTable 1 }	An entry in the list of files, containing information about a single file.
atFilev2InfoFilepath	{ atFilev2InfoEntry 1 }	The full path and name of the file. Files are sorted in alphabetical order and any filepath that is longer than 112 characters will not be displayed due to SNMP Object Identifier length limitations.
atFilev2InfoFileSize	{ atFilev2InfoEntry 2 }	The size of the file in bytes.
atFilev2InfoFileCreationTime	{ atFilev2InfoEntry 3 }	File creation time in the form <MMM DD YYYY HH:MM:SS>. For example, Sep 7 2008 06:07:54.
atFilev2InfoFileIsDirectory	{ atFilev2InfoEntry 4 }	This object will return the value TRUE if the entry is a directory, or FALSE if it is not.
atFilev2InfoFileIsReadable	{ atFilev2InfoEntry 5 }	This object will return the value TRUE if the file is readable, or FALSE if it is not.
atFilev2InfoFileIsWriteable	{ atFilev2InfoEntry 6 }	This object will return the value TRUE if the file is writeable, or FALSE if it is not.
atFilev2InfoFileIsExecutable	{ atFilev2InfoEntry 7 }	This object will return the value TRUE if the file is executable, or FALSE if it is not.
atFilev2USBMediaTable	{ atFilev2 6 }	The USB storage device table, containing information related to USB storage devices.
atFilev2USBMediaEntry	{ atFilev2USBMediaTable 1 }	Data pertaining to an USB storage device instance.
atFilev2USBMediaStackMemberId	{ atFilev2USBMediaEntry 1 }	The index of the stack member hosting this USB media. For devices that are not capable of being stacked, this object will always return the value 1.
atFilev2USBMediaPresence	{ atFilev2USBMediaEntry 2 }	This object indicates whether or not a USB storage device is inserted in a slot. Possible values are: ■ notPresent (1) ■ present (2)

AT-LOG-MIB

The AT Log MIB contains objects for listing log entries from the buffered and permanent logs (Table 64-23). The objects reside in the module log { modules 601 }, organized in the following groups:

- Log Table - objects containing the information from log messages issued by the system, ordered from oldest to newest entry
- Log Options - contains objects used to set up the log options configuration

Table 64-23: Objects defined in AT-LOG-MIB

Object	Object Identifier	Description
log	{ modules 601 }	MIB containing objects for listing log entries from the buffered and permanent logs.
logTable	{ log 1 }	A list of log entries from the source specified in the 'logSource' object. The list is ordered from oldest entry to newest entry. Indexed by: ■ logIndex
logEntry	{ logTable 1 }	Information about a single log entry, from the source specified in the 'logSource' object.
logIndex	{ logEntry 1 }	An index integer. This index is not directly tied to any specific log entry. Over time, the log will grow larger and eventually older entries will be removed from the log.
logDate	{ logEntry 2 }	The date of the log entry. Data resides in the format octet string, in the form YYYY MMM DD, e.g. 2008 Oct 9.
logTime	{ logEntry 3 }	The time of the log entry. Data resides in the format octet string, in the form HH:MM:SS, e.g. 07:15:04.
logFacility	{ logEntry 4 }	The syslog facility that generated the log entry, in the format octet string. See the reference manual for more information.
logSeverity	{ logEntry 5 }	The severity level of the log entry, in the format octet string. Severities are given below: ■ emerg Emergency, system is unusable ■ alert Action must be taken immediately ■ crit Critical conditions ■ errr Error conditions ■ warning Warning conditions ■ notice Normal, but significant, conditions ■ info Informational messages ■ debug Debug-level messages
logProgram	{ logEntry 6 }	The program that generated the log entry, in the format octet string. See the reference manual for more information.
logMessage	{ logEntry 7 }	The message of thew log entry, in the format octet string.
logOptions	{ log 2 }	Contains objects used to set up the required log options configuration.

Table 64-23: Objects defined in AT-LOG-MIB(cont.)

Object	Object Identifier	Description
logSource	{ logOptions 1 }	An integer indicating the source from which the log entries are retrieved. The valid values are: ■ 1 - Buffered log (default) ■ 2 - Permanent log. This information is used when retrieving the logTable objects, and also specifies the log to be cleared when the 'clearLog' object is set.
logAll	{ logOptions 2 }	An integer indicating whether to display all log entries in the logTable objects, or not. The valid values are: ■ 0 - to display only the most recent log messages. This is the default ■ 1 - to show all available log entries. Note: Choosing to display all log entries may result in delays of several seconds when accessing the logTable objects.
clearLog	{ logOptions 3 }	An integer indicating whether to clear the log that is specified by the 'logSource' object. Valid values are: ■ 0 - do not clear log ■ 1 - clear log

AT-IP-MIB

This MIB contains objects for Allied Telesis specific IP address management (Table 64-24). The objects reside in the module atIpMib { modules 602 }.

Table 64-24: Objects defined in AT-IP-MIB

Object	Object Identifier	Description
atIpMib	{ modules 602 }	MIB containing objects for IP addressing management.
AtIpAddressAssignmentType	Textual Convention	Object containing conditional coded values for the IP address assignment type being applied to the interface, referred to by objects in this MIB. The possible values and explanation are: ■ notSet (0) - indicates that the IP address assignment type has not yet been configured. This value can only ever be read. ■ primary (1) - indicates that the address is a primary IP address; only one primary address is allowed per interface. ■ secondary (2) - indicates that the address is a secondary IP address; any number of secondary IP addresses may be applied
AtIpAddressTable	{ atIpMib 1 }	A table containing mappings between primary or secondary IP addresses, and the interfaces they are assigned to. Indexed by: ■ atIpAddressAddrType ■ atIpAddressAddr
AtIpAddressEntry	{ AtIpAddressTable 1 }	Information about the address mapping for a particular interface.
atIpAddressAddrType	{ AtIpAddressEntry 1 }	An indication of the IP version of 'atIpAddressAddr'
atIpAddressAddr	{ AtIpAddressEntry 2 }	The IP address to which this entry's addressing information pertains. The address type of this object is specified in object 'atIpAddressAddrType'.
atIpAddressPrefixLen	{ AtIpAddressEntry 3 }	An integer, specifying the prefix length of the IP address represented by this entry.
atIpAddressLabel	{ AtIpAddressEntry 4 }	The name assigned to the IP address represented by this entry.
atIpAddressIfIndex	{ AtIpAddressEntry 5 }	The index value that uniquely identifies the interface to which this entry is applicable. The interface identified by a particular value of this index corresponds to the interface identified by the same value of the IF-MIB's ifIndex.
atIpAddressAssignmentType	{ AtIpAddressEntry 6 }	The IP address assignment type for this entry (primary or secondary), as described in the Textual Convention 'AtIpAddressAssignmentType'.

Table 64-24: Objects defined in AT-IP-MIB(cont.)

Object	Object Identifier	Description
atIpAddressRowStatus	{ AtIpAddressEntry 7 }	The current status of the IP address entry. The following values may be returned when reading this object: ■ active (1) The IP address is currently mapped to an interface and is valid. ■ notReady (3) The IP address is currently partially configured and is not mapped to an interface. The following values may be written to this object: ■ active (1) An attempt will be made to map the IP address to the configured interface. ■ createAndWait (5) An attempt will be made to create a new IP address entry. ■ destroy (6) The IP address setting will be removed from the device. An entry cannot be made active until its atIpAddressPrefixLen, atIpAddressIfIndex and atIpAddressAssignmentType objects have been set to valid values.

AT-VLAN-MIB

The atVlanStatistics-MIB (Figure 64-3, and Table 64-16) defines objects for managing VLANs. The MIB contains a sub tree for managing VLAN statistics. Objects in the VLAN Statistics sub-tree have the object identifier ({atVlanInfo 1}).

Figure 64-6: The atVlanStatistics MIB tree

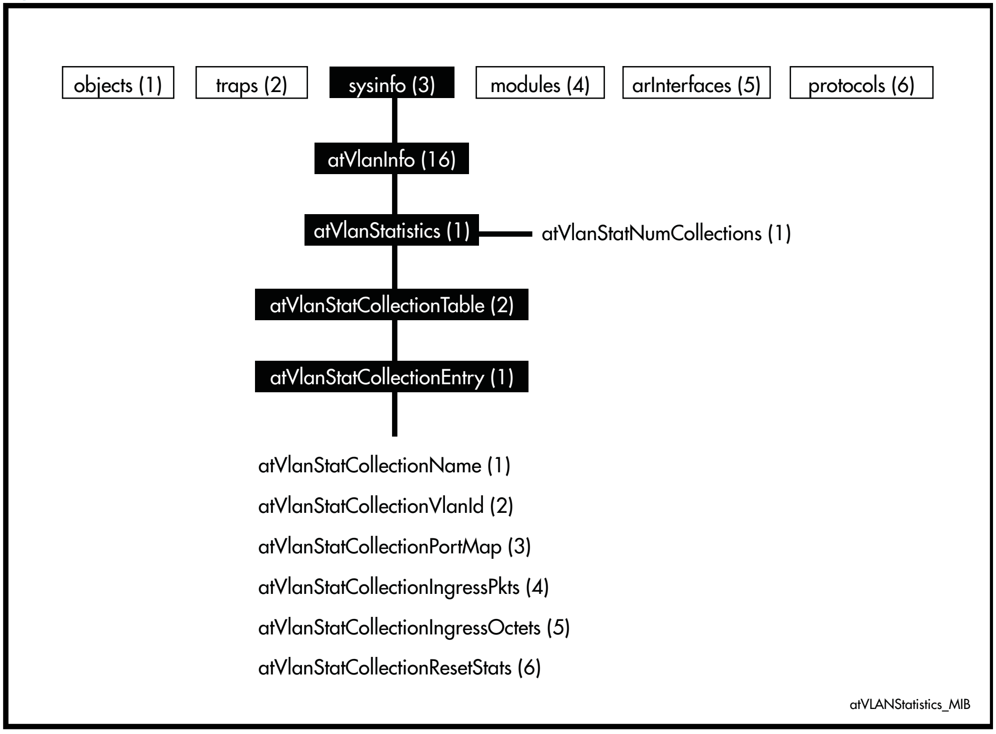

objects (1)
traps (2)
sysinfo (3)
modules (4)
arInterfaces (5)
protocols (6)
atVlanInfo (16)
atVlanStatistics (1)
atVlanStatNumCollections (1)
atVlanStatCollectionTable (2)
atVlanStatCollectionEntry (1)
atVlanStatCollectionName (1)
atVlanStatCollectionVlanId (2)
atVlanStatCollectionPortMap (3)
atVlanStatCollectionIngressPkts (4)
atVlanStatCollectionIngressOctets (5)
atVlanStatCollectionResetStats (6)
atVLANStatistics_MIB

Table 64-25: Objects defined in AT-VLAN-MIB

Object	Object Identifier	Description
atVlanInfo	{ sysinfo 16 }	The VLAN MIB, used for retrieving VLAN specific system data.
atVlanStatistics	{ atVlanInfo 1 }	
atVlanStatNumCollections	{ atVlanStatistics 1 }	The number of unique VLAN statistics gathering instances defined on the device.
atVlanStatCollectionTable	{ atVlanStatistics 2 }	A table of VLAN statistic instances.
atVlanStatCollectionEntry	atVlanStatCollectionTable 1	A series of table entries where each entry represents a unique VLAN statistic gathering instance defined on the device.
atVlanStatCollectionName	atVlanStatCollectionEntry 1	The name of a VLAN statistics collection instance.
atVlanStatCollectionVlanId	atVlanStatCollectionEntry 2	The VLAN ID of ingress packets being monitored by this VLAN statistics collection instance.

Table 64-25: Objects defined in AT-VLAN-MIB(cont.)

Object	Object Identifier	Description
atVlanStatCollectionPortMap	atVlanStatCollectionEntry 3	A bitwise port map indicating the switch ports being monitored by this VLAN statistics collection instance. The bit position within the string, maps to the port with the same index in dot1dBasePortTable in BRIDGE-MIB. A binary '1' indicates that the port is being monitored by this VLAN statistics collection instance, with a '0' indicating that it is not.
atVlanStatCollectionIngressPkts	atVlanStatCollectionEntry 4	The number of ingress packets received and counted by this VLAN statistics collection instance.
atVlanStatCollectionIngressOctets	atVlanStatCollectionEntry 5	The number of octets of data received from ingress packets counted by this VLAN statistics collection instance.
atVlanStatCollectionResetStats	atVlanStatCollectionEntry 6	When read, this object will always return 2 (false). Setting its value to 1 (true) will cause the matching VLAN statistics collection instance's ingress packets and ingress octet values to be reset to zero.

Public MIBs

The following table lists the public MIBs supported by the AlliedWare Plus^{TM} Operating System. In general, all objects are supported except where the relevant protocol or feature is either not supported or not applicable to the device. Any variations from the standard are listed.

Table 64-26: Public MIBs Supported by AlliedWare Plus^{TM}

MIB Name	Reference / Implementation
IANAifType-MIB	www.iana.org/assignments/ianaiftype-mib, IANAifType textual convention.
RFC1155-SMI	RFC 1155, *Structure and Identification of Management Information for TCP/IP-based Internets.*
-	RFC 1212, *Concise MIB Definitions.*
RFC1213-MIB	See IP-MIB.
-	RFC 1215, *A Convention for Defining Traps for use with the SNMP.*
-	RFC 1239, *Reassignment of Experimental MIBs to Standard MIBs.*
IP-MIB	The IP MIB tree encompasses IP-MIB, RFC1213-MIB and IP-FORWARD-MIB definitions. The following documents define the components: ■ RFC 1213, *Management Information Base for Network Management of TCP/IP-based internets: MIB-II* ■ RFC 4292, *IP Forwarding Table MIB* ■ RFC 4293, *Management Information Base for the Internet Protocol (IP)* The following objects **are** supported: ■ ipForwarding ■ ipDefaultTTL ■ All ipAddrTable objects except ipAdEntReasmMaxSize ■ All ipNetToPhysicalTable objects except ipNetToPhysicalRowStatus (all read-only) ■ ipCidrRouteNumber ■ All ipCidrRouteTable objects except ipCidrRouteTos All other objects in these MIBs are not supported. Note that an Enterprise version of ipAddressTable objects is provided by atIpAddressTable in AT-IP-MIB. This provides equivalent functionality along with support for primary and secondary IP addresses.
TCP-MIB	RFC 2012, *SNMPv2 Management Information Base for the Transmission Control Protocol using SMIv2.*
UDP-MIB	RFC 2013, *SNMPv2 Management Information Base for the User Datagram Protocol using SMIv2.*
IP-FORWARD-MIB	See IP-MIB.
-	RFC 2257, *Agent Extensibility (AgentX) Protocol Version 1.*
SNMP-MPD-MIB	RFC 2572, *Message Processing and Dispatching for the Simple Network Management Protocol (SNMP).*
SNMP-COMMUNITY-MIB	RFC 2576, *Coexistence between Version 1, Version 2, and Version 3 of the Internet-standard Network Management Framework.*
SNMPv2-SMI	RFC 2578, *Structure of Management Information Version 2 (SMIv2).*
SNMPv2-TC	RFC 2579, *Textual Conventions for SMIv2.*

Table 64-26: Public MIBs Supported by AlliedWare Plus^{TM}(cont.)

MIB Name	**Reference / Implementation**
SNMPv2-CONF	RFC 2580, *Conformance Statements for SMIv2*.
P-BRIDGE-MIB	RFC 2674, *Definitions of Managed Objects for Bridges with Traffic Classes, Multicast Filtering and Virtual LAN Extensions*. The following objects are not supported: ■ dot1dTpPortOverflowTable ■ dot1dTrafficClassesEnabled ■ dot1dGmrpStatus ■ dot1dPortCapabilitiesTable ■ dot1dUserPriority ■ dot1dTrafficClassPriority ■ dot1dPortOutboundAccessPriorityTable ■ all objects in the dot1dGarp group ■ all objects in the dot1dGmrp group The following read-write object is implemented as read-only: ■ dot1dPortNumTrafficClasses
Q-BRIDGE-MIB	RFC 2674, *Definitions of Managed Objects for Bridges with Traffic Classes, Multicast Filtering and Virtual LAN Extensions*. The following objects are not supported: ■ dot1qGvrpStatus ■ dot1qFdbId ■ dot1qTpFdbAddress ■ dot1qTpGroupTable ■ dot1qForwardAllTable ■ dot1qForwardUnregisteredTable ■ all objects in the dot1qStatic group ■ dot1qVlanTimeMark ■ dot1qVlanIndex ■ dot1qVlanCurrentEgressPorts ■ dot1qVlanCurrentUntaggedPorts ■ dot1qVlanForbiddenEgressPorts ■ dot1qPortGvrpStatus ■ dot1qPortGvrpFailedRegistrations ■ dot1qPortGvrpLastPduOrigin ■ dot1qPortRestrictedVlanRegistration ■ dot1qPortVlanStatisticsTable ■ dot1qPortVlanHCStatisticsTable ■ dot1qLearningConstraintsTable The following read-write objects are implemented as read-only: ■ dot1qPvid ■ dot1qPortAcceptableFrameTypes
VRRP-MIB	RFC 2787, *Definitions of Managed Objects for the Virtual Router Redundancy Protocol.* All objects with read-write and read-create access are implemented as read-only.

Table 64-26: Public MIBs Supported by AlliedWare Plus^{TM}(cont.)

MIB Name	Reference / Implementation
HOST-RESOURCES-MIB	RFC 2790, *Host Resources MIB*. The following objects are not supported: ■ hrStorageAllocationFailures ■ All objects in hrDevice ■ All objects in hrSWRun ■ All objects in hrSWRunPerf ■ All objects in hrSWInstalled ■ All objects in hrMIBAdminInfo
SNMPv2-PDU	RFC 3416, *Version 2 of the Protocol Operations for the Simple Network Management Protocol (SNMP)*.
SNMPv2-TM	RFC 3417, *Transport Mappings for the Simple Network Management Protocol (SNMP)*.
SNMPv2-MIB	RFC 3418, *Management Information Base (MIB) for the Simple Network Management Protocol (SNMP)*.
EtherLike-MIB	RFC 3635, *Definitions of Managed Objects for the Ethernet-like Interface Types*. The following objects have been deprecated: ■ dot3StatsEtherChipSet ■ all objects in the dot3Tests group ■ all objects in the dot3Errors group The following read-write object is implemented as read-only: ■ dot3PauseAdminMode
MAU-MIB	RFC 3636, *Definitions of Managed Objects for IEEE 802.3 Medium Attachment Units (MAUs)*. The following objects are not supported: ■ all objects in the dot3RpMauBasicGroup group ■ ifMauTypeListBits ■ ifMauHCFalseCarriers ■ all object identifiers in the dot3MauType group ■ ifMauAutoNegCapabilityBits ■ ifMauAutoNegCapAdvertisedBits ■ ifMauAutoNegCapReceivedBits ■ ifMauAutoNegRemoteFaultAdvertised ■ ifMauAutoNegRemoteFaultReceived ■ all objects in the mauMod group The following objects have been deprecated: ■ ifMauTypeList ■ all objects in the dot3BroadMauBasicGroup group ■ ifMauAutoNegCapability ■ ifMauAutoNegCapAdvertised ■ ifMauAutoNegCapReceived The following read-write object is implemented as read-only: ■ ifMauStatus
INET-ADDRESS-MIB	RFC 4001, *Textual Conventions for Internet Network Addresses*.
BRIDGE-MIB	RFC 4188, *Definitions of Managed Objects for Bridges*. The following objects are not supported: ■ dot1dStaticTable ■ dot1dBaseDelayExceededDiscards ■ dot1dBasePortMtuExceededDiscards

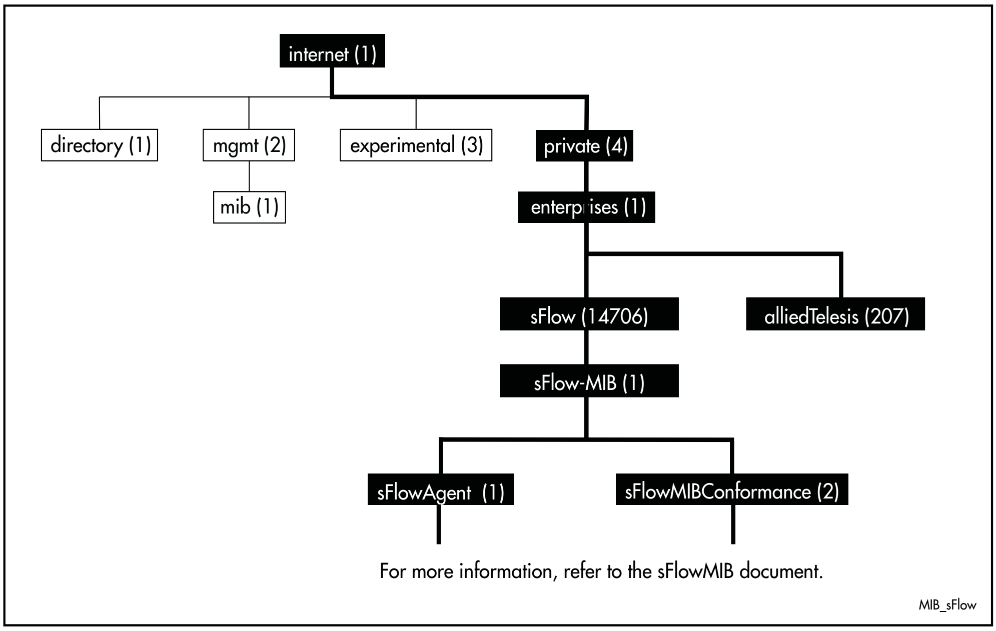
Table 64-26: Public MIBs Supported by AlliedWare Plus^{TM}(cont.)

MIB Name	Reference / Implementation
RSTP-MIB	RFC 4318, *Definitions of Managed Objects for Bridges with Rapid Spanning Tree Protocol.* The following object is deprecated: ■ dot1dStpPathCostDefault
DISMAN-PING-MIB	RFC 4560, *Definitions of Managed Objects for Remote Ping, Traceroute, and Lookup Operations*. The following (lldpLocManAddrTable and lldpConfigManAddrTable) read-write object is implemented as read-only: ■ pingMaxConcurrentRequests You can specify multiple ping operations, but the device only performs one ping at a time (pingMaxConcurrentRequests). The device uses ICMP echo for ping operations (pingImplementationTypeDomains).
LLDP-MIB	*IEEE Standard 802.1AB-2005, Section 12, LLDP MIB Definitions*. The following local management address table supports only a single management address per port: ■ lldpConfigManAddrTable
LLDP-EXT-DOT1-MIB	*IEEE Standard 802.1AB-2005, Annex F, IEEE 802.1 Organizationally Specific TLVs., Section F.7.1, IEEE 802.1LLDP extension MIB module*. In each of the following tables, if one entry is set, all other entries in the table are set to the same value. ■ lldpXdot1ConfigVlanNameTxEnable ■ lldpXdot1ConfigProtoVlanTxEnable ■ lldpXdot1ConfigProtocolTxEnable
LLDP-EXT-DOT3-MIB	*IEEE Standard 802.1AB-2005, Annex G, IEEE 802.3 Organizationally Specific TLVs, Section G.7.1, IEEE 802.3 LLDP extension MIB module*
LLDP-EXT-MED-MIB	*ANSI/TIA-1057- 2006, Section 13.3, LLDP-MED MIB Definition*

Private MIBs

In general, all objects are supported except where the relevant protocol or feature is either not supported or not applicable to the device. The following table lists the private MIBs supported by the AlliedWare Plus^{TM} Operating System. Any variations from the standard are listed.

MIB Name	Reference / Implementation
sFlow-MIB	All MIB objects are fully supported For more information, see www.sflow.org/SFLOW-MIB5.txt



Chapter 65: LLDP Introduction and Configuration

Introduction

This chapter describes the Link Layer Discovery Protocol (LLDP), LLDP for Media Endpoint Devices (LLDP-MED) and Voice VLAN, and general configuration information for these.

LLDP is designed to be managed with the Simple Network Management Protocol (SNMP), and SNMP-based Network Management Systems (NMS). LLDP can be configured, and the information it provides can be accessed, using either the command line interface or SNMP.

- For detailed descriptions of the commands used to configure LLDP and LLDP-MED, see Chapter 66, LLDP Commands.
- For Voice VLAN commands, see Chapter 17, VLAN Commands.
- For information about the LLDP and LLDP-MED MIBs, see "Public MIBs" on page 64.74.

Link Layer Discovery Protocol

Link Layer Discovery Protocol (LLDP) is a Layer 2 protocol defined by the *IEEE Standard 802.1AB-2005*. This switch supports LLDP as specified in this standard, including *Annex F* and *Annex G*.

LLDP enables Ethernet network devices, such as switches and routers, to transmit and/or receive device-related information to or from directly connected devices on the network, and to store such information learned about other devices. The data sent and received by LLDP is useful for many reasons. The switch can discover neighbors—other devices directly connected to it. Devices can use LLDP to advertise some parts of their Layer 2 configuration to their neighbors, enabling some kinds of misconfiguration to be more easily detected and corrected.

LLDP is a link level ("one hop") protocol; LLDP information can only be sent to and received from devices that are directly connected to each other, or connected via a hub or repeater. Advertised information is not forwarded on to other devices on the network.

The information transmitted in LLDP advertisements flows in one direction only, from one device to its neighbors, and the communication ends there. Transmitted advertisements do not solicit responses, and received advertisements do not solicit acknowledgement.

LLDP operates over physical ports (Layer 2) only. For example, it can be configured on switch ports that belong to static or dynamic aggregated links (channel groups), but not on the aggregated links themselves; and on switch ports that belong to VLANs, but not on the VLANs themselves.

LLDP provides a way for the switch to:

- transmit information about itself to neighbors
- receive device information from neighbors
- store and manage information in an LLDP MIB

Each port can be configured to transmit local information, receive neighbor information, or both.

LLDP defines:

- a set of common advertisements ("LLDP Advertisements" on page 65.4)
- a protocol for transmitting and receiving advertisements ("Transmission and Reception" on page 65.8)
- a method for storing the information that is contained within received advertisements ("Storing LLDP Information" on page 65.10)

Interactions

LLDP has the following interactions with other switch features:

- Spanning tree

  Ports blocked by a spanning tree protocol can still transmit and receive LLDP advertisements.
- 802.1x

  Ports blocked by 802.1x port authorization cannot transmit or receive LLDP advertisements. If LLDP has stored information for a neighbor on the port before it was blocked, this information will eventually time out and be discarded.
- VLAN tagging

  LLDP packets are untagged; they do not contain 802.1Q header information with VLAN identifier and priority tagging.
- Virtual Chassis Stacking (VCStack) resiliency link

  When a port is configured as a VCStack resiliency link port, LLDP does not operate on the port; LLDP neither transmits nor receives advertisements, and any LLDP configuration and data stored for the port, including counters, is discarded.
- Mirror ports

  LLDP does not operate on mirror analyzer ports.

LLDP-MED

LLDP for Media Endpoint Devices (LLDP-MED), is an extension of LLDP used between LAN network connectivity devices, such as this switch, and the media endpoint devices connected to them, such as IP phones. LLDP-MED is specified in *ANSI/TIA-1057-2006.* Of the application types specified in *ANSI/TIA-1057-2006*, the switch supports Application Type 1: Voice.

LLDP-MED uses the LLDP advertisement, transmission and storage mechanisms, but transmits, receives, and stores data specifically related to managing the voice endpoint devices. This includes information about network policy, location, hardware configuration, and, for Power over Ethernet-capable devices, power management.

Voice VLAN

Many IP phones (or other IP voice devices) have two interfaces: one to connect to the network and another that allows a computer or similar device to connect to the network via the IP phone. It is often desirable to treat the voice and data traffic separately so that appropriate Quality of Service (QoS) policies can be applied to each. The Voice VLAN feature uses LLDP-MED to convey configuration information (such as VLAN ID and User Priority tagging, and DiffServ Code Point (DSCP)—"Differentiated Services Architecture" on page 35.4) for the voice traffic to the IP phone. In response, the IP phone sends voice traffic according to this configuration. The data traffic coming through the IP phone from the PC is sent with the default configuration, typically untagged with normal priority.

LLDP Advertisements

LLDP transmits advertisements as packets called LLDP Data Units (LLDPDUs). An LLDPDU consists of a set of Type-Length-Value elements (TLV), each of which contains a particular type of information about the device or port transmitting it.

Type-Length-Value (TLV)

A single LLDPDU contains multiple TLVs. TLVs are short information elements that communicate complex data, such as variable length strings, in a standardized format. Each TLV advertises a single type of information, such as its device ID, type, or management addresses. The following table describes fields in a TLV.

Table 65-1: Fields in a Type Length Value element

Field	Description
Type	Identifies the kind of information. It consists of a 7-bit Type code.
Length	Identifies the length of the information. It consists of a 9-bit value that specifies the number of bytes of data in the Value field.
Value	Contains the actual value of the advertised information. This is a variable length data field.

LLDP sends mandatory TLVs in each advertisement; it can also be configured to send one or more optional TLVs, from the following groups:

- Mandatory Base TLVs, included in all LLDP advertisements. See IEEE 802.1AB-2005.
- Optional Base TLVs, which may be included in any LLDP advertisements. See IEEE 802.1AB-2005.
- IEEE 802.1 Organizationally Specific TLVs (802.1 TLVs). See IEEE 802.1AB-2005 Annex F.
- IEEE 802.3 Organizationally Specific TLVs (802.3 TLVs). See IEEE 802.1AB-2005 Annex G.
- LLDP-MED Organizationally Specific TLVs (LLDP-MED TLVs), included in LLDP-MED advertisements. See ANSI/TIA-1057- 2006.

Mandatory and optional TLVs for LLDP and LLDP-MED advertisements are shown in Table 65-2.

Table 65-2: TLVs in LLDP advertisements

TLV	Description
Mandatory Base TLVs—IEEE 802.1AB-2005	
Chassis ID	Identifies the device's chassis. On this switch, this is the MAC address of the switch or stack.
Port ID	Identifies the port that transmitted the LLDPDU.

Table 65-2: TLVs in LLDP advertisements(cont.)

TLV	Description
Time To Live (TTL)	Indicates the length of time in seconds for which the information received in the LLDPDU remains valid. If the value is greater than zero, the information is stored in the LLDP remote system MIB. If the value is zero, the information previously received is no longer valid, and is removed from the MIB.
End of LLDPDU	Signals that there are no more TLVs in the LLDPDU.
Optional Base TLVs—IEEE 802.1AB-2005	
Port description	A description of the device’s port in alpha-numeric format.
System name	The system's assigned name in alpha-numeric format.
System description	A description of the device in alpha-numeric format. This includes information about the device’s hardware and operating system.
System capabilities	The device’s router and bridge functions, and whether or not these functions are currently enabled.
Management address	The address of the local LLDP agent. This can be used to obtain information related to the local device.
IEEE 802.1 Organizationally Specific TLVs (802.1 TLVs)—IEEE 802.1AB-2005 Annex F	
Port VLAN	VLAN identifier that the local port associates with untagged or priority tagged frames.
Port & Protocol VLANs	Whether Port & Protocol VLAN is supported and enabled on the port, and the list of Port & Protocol VLAN identifiers.
VLAN Names	List of VLAN names that the port is assigned to.
Protocol IDs	List of protocols that are accessible through the port, for instance: ■ 9000 (Loopback) ■ 00 26 42 42 03 00 00 00 (STP) ■ 00 27 42 42 03 00 00 02 (RSTP) ■ 00 69 42 42 03 00 00 03 (MSTP) ■ 888e01 (802.1x) ■ aa aa 03 00 e0 2b 00 bb (EPSR) ■ 88090101 (LACP) ■ 00540000e302 (Loop protection) ■ 0800 (IPv4) ■ 0806 (ARP) ■ 86dd (IPv6)
IEEE 802.3 Organizationally Specific TLVs (802.3 TLVs)—IEEE 802.1AB-2005 Annex G	
MAC/PHY Configuration/Status	The current values of the following for the port: ■ Speed and duplex mode auto-negotiation support ■ Auto-negotiation status ■ PMD (physical media dependent) auto-negotiation advertised capability ■ Operational MAU type This TLV is always included in LLDP-MED advertisements.
Power Via MDI	The power-via-MDI capabilities. On devices that are LLDP-MED and PoE-capable, we recommend using the Extended Power-via-MDI TLV instead of this TLV.
Link Aggregation	Whether the link is capable of being aggregated, whether it is currently in an aggregation and if in an aggregation, the port of the aggregation.

Table 65-2: TLVs in LLDP advertisements(cont.)

TLV	Description
Maximum Frame Size	The maximum supported 802.3 frame size that the sending device is capable of receiving—larger frames will be dropped.
LLDP-MED Organizationally Specific TLVs (LLDP-MED TLVs)—ANSI/TIA-1057- 2006	
LLDP-MED Capabilities	Indicates an LLDP-MED capable device, and advertises which LLDP-MED TLVs are supported and enabled, and the device type. For this switch, the device type is Network Connectivity Device. An advertisement containing this TLV is an LLDP-MED advertisement.
Network Policy	Network policy information configured on the port for connected media endpoint devices. The switch supports Application Type 1: Voice, including the following network policy for connected voice devices to use for voice data: ■ Voice VLAN ID ■ Voice VLAN User Priority tagging ■ Voice VLAN Diffserv Code Point (DSCP)
Location Identification	Location information configured for the port, in one or more of the following formats: ■ Civic address ■ Coordinate-based LCI ■ Emergency Location Identification Number (ELIN) For more information, see "LLDP-MED: Location Identification TLV" on page 65.7.
Extended Power-via-MDI	For PoE-capable devices, this TLV includes: ■ Power Type field: Power Sourcing Entity (PSE). ■ Power Source field: current power source, either Primary Power Source or Backup Power Source. ■ Power Priority field: power priority configured on the port. ■ Power Value field: In TLVs transmitted by a Power Sourcing Equipment (PSE) such as this switch, this advertises the power that the port can supply over a maximum length cable based on its current configuration (that is, it takes into account power losses over the cable). In TLVs received from Powered Device (PD) neighbors, the power value is the power the neighbor requests. Available on devices that are PoE-capable.
Inventory Management TLV Set	Includes the following TLVs, based on the current hardware platform and the software version, identical on every port on the switch: ■ Hardware Revision ■ Firmware Revision ■ Software Revision ■ Serial Number ■ Manufacturer Name ■ Model Name ■ Asset ID On Virtual Chassis Stacking devices, the inventory information is based on the current master.

LLDP-MED: Location Identification TLV

Location information can be configured for each port, and advertised to remote devices, which can then transmit this information in calls; the location associated with voice devices is particularly important for emergency call services. All ports may be configured with the location of the switch, or each port may be configured with the location of the remote voice device connected to it.

The location information for a particular port can be configured using one or more of the following three data formats: coordinate-based, Emergency Location Identification Number (ELIN), and civic address. Up to one location of each type can be assigned to a port.

Location configuration information (LCI) in all configured data formats is transmitted in Location Identification TLVs. When LLDP receives a Location Identification TLV, it updates the remote entry in the LLDP-MED MIB with this information.

Co-ordinate LCI

Coordinate-based location data format uses geospatial data, that is, latitude, longitude, and altitude (height or floors), including indications of resolution, with reference to a particular datum: WGS 84, NAD83—North American Vertical Datum of 1988 (NAVD88), or NAD83—Mean Lower Low Water (MLLW). For more information, see *RFC 3825, Dynamic Host Configuration Protocol Option for Coordinate-based Location Configuration Information.*

ELIN LCI

Emergency Location Identification Number (ELIN) location data format provides a unique number for each location for Emergency Call Services (ECS). In North America, ELINs are typically 10 digits long; ELINs up to 25 digits are supported.

Civic Address LCI

The Civic Address location data format uses common street address format, as described in *RFC4776*.

Transmission and Reception

Table 65-3 describes the LLDP transmission and reception processes. Additional LLDP-MED processes are described in "LLDP-MED Operation" on page 65.9.

Table 65-3: LLDP transmission and reception processes

When ...	And ...	Then ...
LLDP is enabled	Ports are configured to transmit LLDP advertisements	Regular LLDP advertisements are sent via these ports at intervals determined by the transmit interval. Each advertisement contains local information (from the Local Systems MIB) for all the mandatory TLVs and the optional TLVs that the port is configured to send.
	Ports are configured to receive LLDP advertisements	Information received in advertisements via these ports is stored in the Neighbor table (Remote Systems MIB). This information is retained until it is replaced by a more recent advertisement from the same neighbor or it times out (the TTL elapses).
Local information changes	The transmission delay time has elapsed since the last advertisement was transmitted	New advertisements are sent containing the new set of local information.
Neighbor information changes	Notifications are enabled, and the notification interval has elapsed since the last notification was sent	The SNMP notification (trap) lldpRemTablesChange is sent.
LLDP transmission and reception is disabled on a port.	An LLDP command was used to do this	It transmits a final 'shutdown' LLDPDU with a Time-To-Live (TTL) TLV that has a value of "0". This tells any remote neighboring devices to remove the information associated with this switch from their remote systems MIB. Then it stops transmitting and receiving advertisements. The neighbor information remains in the Remote Systems MIB until it times out.
	A shutdown command was used on the port	It makes a best effort to send a shutdown LLDPDU. Then it stops transmitting and receiving advertisements. The neighbor information remains in the Remote Systems MIB until it times out.
	Something else disabled LLDP, such as Virtual Chassis Stacking (VCStack) failover	It does not send a shutdown LLDPDU. It stops transmitting and receiving advertisements. The neighbor information remains in the Remote Systems MIB until it times out.
	It is enabled again	LLDP reinitializes and resumes transmitting and receiving advertisements after the reinitialization interval has elapsed.
The Neighbor table has 1600 neighbors		It discards any further neighbors.
LLDP receives a LLDPDU or TLV with a detectable error		It discards the incorrect TLV.
LLDP receives a TLV it does not recognize	It contains no basic format errors	It stores it for possible later retrieval by network management (in the unrecognized TLV information table lldpRemUnknownTLVTable in the LLDP MIB).

LLDP-MED Operation

When LLDP is enabled, LLDP-MED is enabled by default, and uses the same LLDP transmission and reception process described in Table 65-3. When LLDP receives an advertisement indicating a newly connected LLDP-MED-capable device on a port, it transmits one LLDP-MED advertisement per second via this port, a configurable number of times (the *fast start count*). Thereafter, it sends regular advertisements at the LLDP transmit interval. When the last advertisement for an LLDP-MED-capable device connected to the port times out, it stops sending LLDP-MED advertisements via the port.

If LLDP-MED notifications are enabled for a port, and SNMP traps for LLDP are enabled, LLDP-MED generates a *Topology Change Notification (LLDP-MED lldpXMedTopology ChangeDetected)* when a new LLDP-MED compliant IP telephony device is connected to a port or removed from a port. This notification includes the following information:

- IP Phone Chassis ID and Chassis ID sub-type (IP address)
- LLDP Endpoint Device Class
- Switch Chassis ID (MAC address) and Port ID where the device is attached.

Storing LLDP Information

When an LLDP device receives a valid LLDP advertisement from a neighboring network device, it stores the information in an IEEE-defined Simple Network Management Protocol (SNMP) Management Information Base (MIB).

LLDP stores information in the LLDP MIB defined in Section 12 of the *IEEE Standard 802.1AB-2005*, its extensions defined in *Annex F*, *Annex G*, and ANSI/TIA-1057- 2006, about:

LLDP-EXT-MED-MIB ANSI/TIA-1057- 2006, Section 13.3, LLDP-MED MIB Definition

- Local system information. This is the information that LLDP can transmit in advertisements to its neighbors.
- Remote systems information. This is the data that the device receives in advertisements from its neighbors.
- LLDP configuration. This can be used with SNMP to configure LLDP on the device.
- LLDP statistics. This includes information about LLDP operation on the device, including packet and event counters.

This information can be accessed either via SNMP, or directly using the command line interface.

Local system

Information about your device is called local system information. The LLDP local system MIB maintains this information, which consists of device details, as well as any user-configured information that you have set up for your switch, for example a port description or a management address.

LLDP on this device can store one management address per port, and transmit this in LLDP advertisements. It can store multiple management addresses received from each neighbor.

Remote systems

Information gained from neighboring devices is called remote system information. The LLDP remote systems MIB maintains this information.

The length of time for which neighbor information remains in the LLDP remote systems MIB is determined by the Time-To-Live (TTL) value of received LLDPDUs. When it receives an advertisement from a neighbor, LLDP starts a timer based on the Time To Live (TTL) information in the advertisement. The Time To Live (TTL) information in an advertisement is: TTL=transmit interval x holdtime multiplier. If the TTL elapses, for instance if the neighbor has been removed, LLDP deletes the neighbor's information from the MIB. This ensures that only valid LLDP information is stored.

Whenever a new neighbor is discovered, or an existing neighbor sends an advertisement with new information that differs from the previous advertisement, for example a new or changed TLV, a remote tables change event is activated. If SNMP notifications are enabled, the notification lldpRemTablesChange is sent.

To prevent the remote systems MIB from using large amounts of memory and possibly affecting the operation of your switch, it limits the number of neighbors it stores information for to 1600. If it is storing information from 1600 neighbors, and detects any more neighbors, it is considered to have too many neighbors, and discards advertisements from the rest. There is no per-port limit to the number of neighbors.

SNMP utilities

An SNMP utility can read the Neighbors table MIB (Remote Systems Data in the LLDP MIB) on a device to find out about the LLDP neighbors it is directly connected to on each port. Then it can read the Neighbors table MIB on each of these neighbors to find out about their neighboring LLDP devices, and so on.

Configuring LLDP

You can configure LLDP on the device using either:

- the command line interface. For detailed descriptions of the commands, see Chapter 66, LLDP Commands, or
- SNMP—see Chapter 64, SNMP MIBs.

This section includes the following command line interface configuration procedures:

- "Configure LLDP" on page 65.12— This procedure includes configuration for LLDP between network connectivity devices; it does not include LLDP-MED. If you are configuring LLDP-MED only, use the following procedure instead of this one.
- "Configure LLDP-MED" on page 65.14—This procedure includes the LLDP configuration required to support LLDP-MED, as well as specific LLDP-MED and Voice VLAN configuration.
- "Configure Authentication for Voice VLAN" on page 65.19—This procedure includes 802.1X port authentication configuration including dynamic VLAN assignment to be used with LLDP-MED. Use the previous procedure before using this one.

Because LLDP is often used together with SNMP, consider configuring SNMP before you configure LLDP. LLDP transmits large amounts of data about the network. For security reasons, we recommend configuring SNMP for SNMP version 3 only (for read and write access). Remove all SNMPv1 and SNMPv2 configuration. See Chapter 62, SNMP Introduction, and Chapter 63, SNMP Commands.

Configure LLDP

Use the procedure in Table 65-4 below to configure LLDP.

Some optional TLVs send information that can be configured by other commands. If LLDP will be configured to send these TLVs, consider whether to configure the corresponding parameters first.

- Port Description. See the description (interface) command on page 12.2.
- System Name. See the hostname command on page 8.14.

Table 65-4: Configuration procedure for LLDP

Enable LLDP		
1.	`awplus#configure terminal`	Enter Configuration mode.
2.	`awplus(config)#lldp run`	Enable LLDP.
Configure ports for LLDP Configure each port to determine whether and which LLDP messages are transmitted and received. If all the ports running LLDP require the same configuration, configure them all together. Otherwise repeat these commands for each port or group of ports that requires a particular configuration.		
3.	`awplus(config)# interface <port-list>`	Enter Interface Configuration mode for the switch ports.

4.	`awplus(config-if)#lldp tlv-select {[<tlv>]...}`
	`awplus(config-if)#lldp tlv-select all`
		By default, the mandatory TLVs are included in LLDP messages. Enable the transmission of one or more optional TLVs through these port as required.

5.	`awplus(config-if)#exit`	Return to Global Configuration mode.
6.	`awplus(config)#interface <port-list>`	By default, transmission and reception of LLDP advertisements is enabled on all ports. Enter Interface Configuration mode for any switch ports that should have transmission or reception disabled.
7.	`awplus(config-if)#no lldp {[transmit] [receive]}`	Disable transmission and/or reception as required.
8.	`awplus(config-if)#exit`	Return to Global Configuration mode.
9.	`awplus(config)#exit`	Return to Privileged Exec mode.
Check LLDP configuration		

10.	`awplus#show lldp`
	`awplus#show lldp interface [<port-list>]`
	`awplus#show lldp local-info [base] [dot1] [dot3] [med] [interface <port-list>]`
	`awplus#show running-config lldp`
		Review the LLDP configuration.

Monitor LLDP		

11.	`awplus#show lldp neighbors`
	`awplus#show lldp neighbors detail`
	`awplus#show lldp statistics`
	`awplus#show lldp statistics interface [<port-list>]`
		Monitor LLDP operations and display neighbor information as required.

Table 65-4: Configuration procedure for LLDP**(cont.)**

Advanced LLDP configuration

The configuration procedure above and the defaults for other settings suit most networks. Use the following commands for fine tuning if necessary.

Timer intervals should be long enough not to create unnecessarily high numbers of advertisements when there are topology changes. However, be aware that if the intervals are long, a neighbor's information can continue to be stored after its information has changed, or after it is disconnected.

12.	awplus#configure terminal	Enter Configuration mode.
13.	awplus(config)#interface *<port-list>*	Enter Interface Configuration mode for the switch ports.
14.	awplus(config-if)#lldp management-address <ipaddr>	Override the default LLDP management address advertised through this port if required. This must be an IPv4 address that is already configured on the device. To see the management address that will be advertised, use the show lldp local-info command on page 66.40.
15.	awplus(config-if)#lldp notifications	By default, SNMP notifications are not transmitted. Enable them for these ports if required. (SNMP LLDP traps (notifications) must also be enabled.)
16.	awplus(config-if)#exit	Return to Global Configuration mode.
17.	awplus(config)#lldp timer <5-32768>	The transmit interval determines how often regular LLDP transmits advertisements from each port. The transmit interval must be at least four times the transmission delay. Default: 30 seconds
18.	awplus(config)#lldp notification-interval <5-3600>	The notification interval determines the minimum interval between sending SNMP notifications (traps). Default: 5 seconds
19.	awplus(config)#lldp tx-delay <1-8192>	A series of successive changes over a short period of time can trigger the agent to send a large number of LLDPDUs. To prevent this, there is a transmission delay timer. This establishes a minimum length of time that must elapse between successive LLDP transmissions. The transmission delay cannot be greater than a quarter of the transmit interval. Default: **2** seconds
20.	awplus(config)#lldp reinit <1-10>	Reinitialization delay timer determines the minimum time after disabling LLDP on a port before it can reinitialize. Default: **2** seconds
21.	awplus(config)#lldp holdtime-multiplier <2-10>	The transmit interval is multiplied by the holdtime multiplier to give the Time To Live (TTL) value that is advertised to neighbors. Default: **4**
22.	awplus(config)#exit	Return to Privileged Exec mode.

Clear data

If necessary, you can clear either neighbor information or LLDP statistics for particular ports or all ports.

23.	awplus#clear lldp table [interface <port-list>]	Clear the information from the table of neighbor information.
24.	awplus#clear lldp statistics [interface <port-list>]	Clear LLDP statistics (packet and event counters).

Configure LLDP-MED

Use the procedure in Table 65-5 to configure LLDP-MED and Voice VLAN for voice devices connected to the switch.

Consider whether you also need to configure:

- Simple Network Management Protocol (Chapter 63, SNMP Commands)
- 802.1X port authentication (Chapter 38, 802.1X Commands, Chapter 40, Authentication Commands, Chapter 42, AAA Commands)
- RADIUS server (Chapter 48, Local RADIUS Server Commands, or Chapter 44, RADIUS Commands)
- Quality of Service (Chapter 36, QoS Commands)
- Access Control Lists (Chapter 32, IPv4 Hardware Access Control List (ACL) Commands and Chapter 33, IPv4 Software Access Control List (ACL) Commands)
- Power over Ethernet (PoE), if the switch supports PoE (Chapter 23, Power over Ethernet Commands)

In most cases, configuring LLDP-MED using SNMP or using the CLI command line interface (CLI) described in Chapter 66, LLDP Commands has the same effect. However, the effect of configuring location information using SNMP differs from the CLI. When location information is assigned to a port by SNMP and a matching location is not found on the device, then a new location is automatically created and assigned to the specified port. If the location is unset by SNMP later, then the location is removed to prevent accumulating SNMP-set location information. However, if the location is being used for other ports, the automatically created location is not removed until no ports use it. Once it is modified or assigned to other ports by CLI commands, the location remains even after no ports use the location.

Table 65-5: Configuration procedure for Voice VLAN and LLDP-MED

Configure a Voice VLAN Create a VLAN for voice data from voice endpoint devices connected to ports on the switch. Specify the network policy for voice data in this voice VLAN. LLDP-MED sends the network policy to voice devices connected to these ports. The voice devices use this network policy to determine the VLAN, priority and DSCP tagging of voice data it transmits.		
1.	`awplus#` configure terminal	Enter Global Configuration mode.
2.	`awplus(config)#` vlan database	Enter VLAN Database Configuration mode.
3.	`awplus(config-vlan)#` vlan <vid> [name <vlan-name>] [state {enable|disable}]	Create a VLAN to be used for the voice data to and from voice devices connected to the switch. By default, the new VLAN is enabled.
4.	`awplus(config-vlan)#` exit	Return to global configuration mode.
5.	`awplus(config)# interface <port-list>`	Enter interface configuration mode for the ports to be configured with the same network policy. This may be all the switch ports with voice devices connected to them, or a subset if the network policy will differ between ports.

Table 65-5: Configuration procedure for Voice VLAN and LLDP-MED(cont.)

6.	awplus(config-if)# switchport voice vlan [<vid>|dot1p|dynamic|untagged]	Specify the VLAN tagging to be used for voice data on these ports. Use the **dynamic** option if the VLAN tagging will be allocated dynamically by a RADIUS server. To configure authentication and dynamic VLAN allocation using the local RADIUS server, see the procedure in Table 65-6 on page 65.19. Default: **none**.
7.	awplus(config-if)# switchport voice vlan priority <0-7>	Specify the priority-tagging that voice endpoint devices should put into their data packets. Default: **5**.
8.	awplus(config-if)# switchport voice dscp <0-63>	Specify the DSCP value that voice endpoint devices should put into their data packets. Default: **0**.
9.	awplus(config-if)# exit	Return to global configuration mode.
Enable LLDP		
10.	awplus(config)# lldp run	Enable LLDP on the switch. Default: LLDP is disabled.
11.	awplus(config)# interface *<port-list>*	Enter interface configuration mode for the switch ports LLDP is NOT to run on.
12.	awplus(config-if)# no lldp {[transmit] [receive]}	Disable transmission or reception on these ports as required. Default: **transmit** and **receive** enabled.
13.	awplus(config-if)# exit	Return to global configuration mode.
Configure LLDP-MED location information Create civic address, coordinate, and/or ELIN locations, and assign them to switch ports.		
14.	awplus(config)# location civic-location identifier <civic-loc-id>	Specify a civic location ID, and enter configuration mode for this identifier.
15.	awplus(config-civic)# country <country> awplus(config-civic)# city <city> awplus(config-civic)# primary-road-name <primary-road-name> awplus(config-civic)# street-suffix <street-suffix> awplus(config-civic)# house-number <house-number> awplus(config-civic)# *<other-civic-location-parameters ...>*	Specify the civic address location information for the civic address location ID. You must specify a country first, using the upper-case two-letter country code, and then at least one more parameter. For the full set of parameters you can use to specify civic address location, see the location civic-location configuration command on page 66.23.
16.	awplus(config-civic)# exit	Return to global configuration mode.
17.	awplus(config)# location coord-location identifier <coord-loc-id>	Specify a coordinate location identifier, and enter configuration mode for this identifier.

Table 65-5: Configuration procedure for Voice VLAN and LLDP-MED(cont.)

18.	awplus(config-coord)# latitude <latitude> awplus(config-coord)# lat-resolution <lat-resolution> awplus(config-coord)# longitude <longitude> awplus(config-coord)# long-resolution <long-resolution> awplus(config-coord)# altitude <altitude> {meters|floor} awplus(config-coord)# alt-resolution <alt-resolution> awplus(config-coord)# datum {wgs84|nad83-navd|nad83-mllw}	Specify the coordinate location for the coordinate location identifier.
19.	awplus(config-coord)# exit	Return to global configuration mode.
20.	awplus(config)# location elin-location <elin> identifier <elin-loc-id>	Specify an ELIN location identifier, and the ELIN for this identifier.
21.	awplus(config)# interface *<port-list>*	Enter interface configuration mode for one or more switch ports which require the same location information.
22.	awplus(config-if)# location civic-location-id <civic-loc-id> awplus(config-if)# location coord-location-id <coord-loc-id> awplus(config-if)# location elin-location-id <elin-loc-id>	Assign the civic, coordinate, and/or ELIN location identifier to these ports. LLDP-MED will send the location information associated with a port to the voice endpoint device attached to it.
23.	awplus(config-if)# exit	Return to global configuration mode.
24.	awplus(config)# exit	Return to Privileged Exec mode.
Review the LLDP configuration		
25.	awplus# show lldp	Check general LLDP configuration settings.
26.	awplus# show lldp interface [<port-list>]	Check LLDP configuration for ports.
27.	awplus# show lldp local-info [base] [dot1] [dot3] [med] [interface <port-list>]	Check the information that may be transmitted in LLDP advertisements from ports.

28.	awplus# show location {civic-location|coord-location|elin-location}
	awplus# show location {civic-location|coord-location|elin-location} identifier {<civic-loc-id>|<coord-loc-id>|<elin-loc-id>}
	awplus# show location {civic-location|coord-location|elin-location} interface <port-list>
		Check the location information.

29.	awplus# show running-config lldp	If you want to display all the LLDP configuration, use this command.
Monitor LLDP-MED		
30.	awplus# show lldp neighbors [interface <port-list>] awplus# show lldp neighbors detail [base] [dot1] [dot3] [med] [interface <port-list>] awplus# show lldp statistics awplus# show lldp statistics interface [<port-list>]	Monitor LLDP operation.

Table 65-5: Configuration procedure for Voice VLAN and LLDP-MED(cont.)

Advanced configuration

The configuration procedure above and the defaults for other settings suit most networks. Use the following commands for fine tuning if necessary. For information about other advanced configuration for LLDP, including LLDP timers, see Table 65-4.

31.	awplus#configure terminal	Enter Global Configuration mode.
32.	awplus(config)# lldp faststart-count <1-10>	By default, when LLDP-MED detects an LLDP-MED capable device on a port, it sends 3 advertisements at 1s intervals. Change the fast start count if required. Default: fast start count is 3
33.	awplus(config)# lldp non-strict-med-tlv-order-check	By default non-strict order checking for LLDP-MED advertisements is disabled. That is, strict order checking is applied to LLDP-MED advertisements, and LLDP-MED TLVs in non-standard order are discarded. If you require LLDP-MED advertisements with non-standard TLV order to be received and stored, enable non-strict order checking.
34.	awplus(config)# interface *<port-list>*	Enter interface configuration mode for switch ports which will have the same advanced configuration.
35.	awplus(config-if)# lldp management-address <ipaddr>	Override the default LLDP management address advertised through this port if required. This must be an IPv4 address that is already configured on the device. To see the management address that will be advertised, use the show lldp local-info command on page 66.40.
36.	awplus(config-if)# lldp med-notifications	By default, SNMP notifications are not transmitted. Enable LLDP-MED Topology Change Detected notifications for these ports if required. (SNMP LLDP traps (notifications) must also be enabled.) Default: LLDP-MED notifications disabled
37.	awplus(config-if)# lldp tlv-select {[<tlv>]...}	Enable the transmission of one or more optional LLDP TLVs in LLDP-MED advertisements through this port as required. The **mac-phy-config** TLV is transmitted in LLDP-MED advertisements whether or not it is enabled by this command. Default: all mandatory TLVs are enabled.

38.	awplus(config-if)# lldp med-tlv-select {[capabilities] [network-policy] [location] [power-management-ext] [inventory-management]}
	awplus(config-if)# lldp med-tlv-select all
	awplus(config-if)# no lldp med-tlv-select {[capabilities] [network-policy] [location] [power-management-ext] [inventory-management]}
	awplus(config-if)# no lldp med-tlv-select all
		Enable or disable the transmission of optional LLDP-MED TLVs in LLDP-MED advertisements through these ports as required.
		Default: **capabilities**, **network-policy**, **location**, **power-management** are enabled.

39.	awplus(config-if)# exit	Return to global configuration mode.

Table 65-5: Configuration procedure for Voice VLAN and LLDP-MED(cont.)

40.	`awplus(config)#` exit	Return to privileged exec mode.
Clear data If necessary, you can clear either neighbor information or LLDP statistics for particular ports or all ports.		
41.	`awplus#` clear lldp table [interface <port-list>]	Clear the information from the table of neighbor information.
42.	`awplus#` clear lldp statistics [interface <port-list>]	Clear LLDP statistics (packet and event counters).

Configure Authentication for Voice VLAN

Use the following procedure with LLDP-MED and Voice VLAN to configure 802.1X port authentication and dynamic VLAN assignment using the local RADIUS server on the switch to which the voice endpoint devices are connected.

This procedure assumes that you have already:

- configured Voice VLAN and LLDP-MED using the procedure in Table 65-5 on page 65.14
- set switchport voice vlan to **dynamic** in the above procedure

This procedure configures the local RADIUS server. If your configuration uses one or more remote RADIUS servers instead, set the IP addresses of the remote RADIUS servers using the radius-server host command (Step 3 on page 19), and skip all the steps that configure the local RADIUS server (Step 3 on page 19 to Step 14 on page 20).

Table 65-6: Configuration procedure for Voice VLAN with RADIUS authentication and dynamic VLAN

Configure the IP address of the RADIUS host.		
1.	`awplus#`configure terminal	Enter Global Configuration mode.
2.	`awplus(config)#`radius-server host 127.0.0.1 key *<key-string>*	Configure the IP address for the RADIUS server to be the local loopback interface address, so that RADIUS requests are sent to the local RADIUS server. Set the key that Network Access Servers (NAS) will need to use to get access to this RADIUS server. RADIUS server hosts configured using this command are included in the default RADIUS server group.
Enable the local RADIUS server.		
3.	`awplus(config)#` radius-server local	Enter RADIUS Server Configuration mode.
4.	`awplus(config-radsrv)#` server enable	Enable the local RADIUS server.
5.	`awplus(config-radsrv)#` nas 127.0.0.1 key *<key-string>*	Set the switch as a client device (Network Access Server), to allow it to send authentication requests to the local RADIUS server. Use the same local loopback interface IP address and key as in the radius-server host command used in Step 2 on page 19.
Configure a local RADIUS user group for connected PCs.		
6.	`awplus(config-radsrv)#` group <user-group-name>	Create a local RADIUS server user group for PCs connected to the switch, and enter RADIUS Server Group Configuration mode.
7.	`awplus(config-radsrv-group)#` vlan {<vid>|<vlan-name>}	Set the VLAN ID for the user group. This will assign the untagged VLAN ID to authenticated ports for PCs connected to the switch. To create multiple user groups for PCs with different VLANs, repeat these two steps.
8.	`awplus(config-radsrv-group)#`exit	Return to RADIUS Server Configuration mode.
Configure a local RADIUS user group for connected phones.		
9.	`awplus(config-radsrv)#` group <user-group-name>	Create a new local RADIUS server user group for phones connected to the switch, and enter RADIUS Server Group Configuration mode.

Table 65-6: Configuration procedure for Voice VLAN with RADIUS authentication and dynamic VLAN(cont.)

10.	awplus(config-radsrv-group)# vlan {<vid>|<vlan-name>}	Configure the local RADIUS user group for connected phones to use the same VLAN as the PCs in Step 7, so that the phones have access to the same untagged VLAN as the PCs.
11.	awplus(config-radsrv-group)# egress-vlan-id <*vid*> tagged	Set the Egress-VLAN ID attribute for the user group, and set it to send tagged frames. This will assign the tagged VLAN ID to authenticated ports for phones connected to the switch. To create multiple user groups for phones with different VLANs, repeat these two steps.
12.	awplus(config-radsrv-group)# exit	Return to RADIUS Server Configuration mode.
Add users to the local RADIUS server.		
13.	awplus(config-radsrv)# user <radius-user-name> password <user-password> group <user-group>	Add RADIUS user names and passwords to the local RADIUS server for authenticating PCs and phones. Assign the corresponding RADIUS server user groups configured in Step 6 and Step 9. See the user (RADIUS server) command on page 48.36.
14.	awplus(config-radsrv)# exit	Return to Global Configuration mode.
Create VLANs.		
15.	awplus(config)# vlan database	Enter VLAN Database Configuration mode.
16.	awplus(config-vlan)# vlan <*vid-range*>	Create the VLANs corresponding to the VLAN IDs that will be allocated to the authenticated ports, as configured in Step 7, Step 10, and Step 11.
17.	awplus(config-vlan)# exit	Return to Global Configuration mode.
Configure 802.1X port authentication.		
18.	awplus(config)# aaa authentication dot1x default group radius	Enable 802.1X port authentication and set it to use the default group of RADIUS servers that contains all RADIUS server hosts configured using the radius-server host command—in this procedure, the default group consists of the local RADIUS server.
19.	awplus(config)# interface *<port-list>*	Enter interface configuration mode for the ports that have users (PCs and phones) connected to them.
20.	awplus(config-if)# dot1x port-control auto	Enable 802.1X for port authentication on these ports.
21.	awplus(config-if)# auth host-mode multi-supplicant	Configure the ports to use multi-supplicant mode for authentication, so that the phone and PC can be dynamically allocated to different VLANs.
22.	awplus(config-if)# auth dynamic-vlan-creation	Configure the ports to accept dynamic VLAN allocation. In this procedure, the RADIUS server user groups for both the PCs and the phones use the same VLAN (Step 7 and Step 10), so the default rule (**deny**) allows them both the access they need to the port VLAN. For other options, see the auth dynamic-vlan-creation command on page 40.6. Default: **deny** differently assigned VLAN IDs.
23.	awplus(config-if)# exit	Return to Global Configuration mode.
24.	awplus(config)# exit	Return to Privileged Exec mode.

Table 65-6: Configuration procedure for Voice VLAN with RADIUS authentication and dynamic VLAN(cont.)

Review the authentication configuration.		
25.	awplus# show radius local-server group [<user-group-name>] awplus# show radius local-server nas [<ip-address>] awplus# show radius local-server user [<user-name>]	Check the local RADIUS server configuration.
26.	awplus# show vlan {all|brief|dynamic|static|<1-4094>}	Check the VLAN configuration.
27.	awplus# show dot1x [all]	Check the 802.1X authentication configuration.

Chapter 66: LLDP Commands

Introduction

LLDP and LLDP-MED can be configured using the commands in this chapter, or by using SNMP with the LLDP-MIB and LLDP-EXT-DOT1-MIB ("Public MIBs" on page 64.74). The Voice VLAN feature can be configured using commands in Chapter 17, VLAN Commands. For more information about LLDP, see Chapter 65, LLDP Introduction and Configuration.

LLDP can transmit a lot of data about the network. Typically, the network information gathered using LLDP is transferred to a Network Management System by SNMP. For security reasons, we recommend using SNMPv3 for this purpose (Chapter 62, SNMP Introduction, Chapter 63, SNMP Commands).

LLDP operates over physical ports only. For example, it can be configured on switch ports that belong to static or dynamic channel groups, but not on the channel groups themselves.

Command List

This chapter contains an alphabetical list of commands used to configure LLDP.

clear lldp statistics

This command clears all LLDP statistics (packet and event counters) associated with specified ports. If no port list is supplied, LLDP statistics for all ports are cleared.

Syntax

```
clear lldp statistics [interface <port-list>]
```

Parameter	Description
<port-list>	The ports for which the statistics are to be cleared.

Mode Privileged Exec

Examples To clear the LLDP statistics on ports 1.0.1 and 1.0.7, use the command:

```
awplus# clear lldp statistics interface port1.0.1,port1.0.7
```

To clear all LLDP statistics for all ports, use the command:

```
awplus# clear lldp statistics
```

Related Commands show lldp statistics
show lldp statistics interface

clear lldp table

This command clears the table of LLDP information received from neighbors through specified ports. If no port list is supplied, neighbor information is cleared for all ports.

Syntax

```
clear lldp table [interface <port-list>]
```

Parameter	Description
<port-list>	The ports for which the neighbor information table is to be cleared.

Mode Privileged Exec

Examples To clear the table of neighbor information received on ports 1.0.1 and 1.0.7, use the command:

```
awplus# clear lldp table interface port1.0.1,port1.0.7
```

To clear the entire table of neighbor information received through all ports, use the command:

```
awplus# clear lldp table
```

Related Commands show lldp neighbors

debug lldp

This command enables specific LLDP debug for specified ports. When LLDP debugging is enabled, diagnostic messages are entered into the system log. If no port list is supplied, the specified debugging is enabled for all ports.

The **no** variant of this command disables specific LLDP debug for specified ports. If no port list is supplied, the specified debugging is disabled for all ports.

Syntax

```
debug lldp {[rx][rxpkt][tx][txpkt]} [interface [<port-list>]]

debug lldp operation

no debug lldp {[rx][rxpkt][tx][txpkt]} [interface [<port-list>]]

no debug lldp operation

no debug lldp all
```

Parameter	Description
`rx`	LLDP receive debug.
`rxpkt`	Raw LLDPDUs received in hex format.
`tx`	LLDP transmit debug.
`txpkt`	Raw Tx LLDPDUs transmitted in hex format.
<port-list>	The ports for which debug is to be configured.
`operation`	Debug for LLDP internal operation on the switch.
`all`	Disables all LLDP debugging for all ports.

Default By default no debug is enabled for any ports.

Mode Privileged Exec

Examples To enable debugging of LLDP receive on ports 1.0.1 and 1.0.7, use the command:

```
awplus# debug lldp rx interface port1.0.1,port1.0.7
```

To enable debugging of LLDP transmit with packet dump on all ports, use the command:

```
awplus# debug lldp tx txpkt
```

To disable debugging of LLDP receive on ports 1.0.1 and 1.0.7, use the command:

```
awplus# no debug lldp rx interface port1.0.1,port1.0.7
```

To turn off all LLDP debugging on all ports, use the command:

```
awplus# no debug lldp all
```

Related Commands

show debugging lldp
show running-config lldp
terminal monitor

lldp faststart-count

Use this command to set the fast start count for LLDP-MED. The fast start count determines how many fast start advertisements LLDP sends from a port when it starts sending LLDP-MED advertisements from the port, for instance, when it detects a new LLDP-MED capable device.

The **no** variant of this command resets the LLDPD-MED fast start count to the default (3).

Syntax

```
lldp faststart-count <1-10>
```

```
no lldp faststart-count
```

Parameter	Description
<1-10>	The number of fast start advertisements to send.

Default The default fast start count is 3.

Mode Global Configuration

Examples To set the fast start count to 5, use the command:

```
        awplus# configure terminal
awplus(config)# lldp faststart-count 5
```

To reset the fast start count to the default setting (3), use the command:

```
        awplus# configure terminal
awplus(config)# no lldp faststart-count
```

Related Commands show lldp

lldp holdtime-multiplier

This command sets the holdtime multiplier value. The transmit interval is multiplied by the holdtime multiplier to give the Time To Live (TTL) value that is advertised to neighbors.

The **no** variant of this command sets the multiplier back to its default.

Syntax

```
lldp holdtime-multiplier <2-10>
```

```
no lldp holdtime-multiplier
```

Parameter	Description
<2-10>	The multiplier factor.

Default The default holdtime multiplier value is 4.

Mode Global Configuration

Usage The Time-To-Live defines the period for which the information advertised to the neighbor is valid. If the Time-To-Live expires before the neighbor receives another update of the information, then the neighbor discards the information from its database.

Examples To set the holdtime multiplier to 2, use the commands:

```
        awplus# configure terminal
awplus(config)# lldp holdtime-multiplier 2
```

To set the holdtime multiplier back to its default, use the commands:

```
        awplus# configure terminal
awplus(config)# no lldp holdtime-multiplier 2
```

Related Commands show lldp

lldp management-address

This command sets the IPv4 address to be advertised to neighbors (in the Management Address TLV) via the specified ports. This address will override the default address for these ports.

The **no** variant of this command clears the user-configured management IP address advertised to neighbors via the specified ports. The advertised address reverts to the default.

Syntax

```
lldp management-address <ipaddr>

no lldp management-address
```

Parameter	Description
<ipaddr>	The IPv4 address to be advertised to neighbors, in dotted decimal format. This must be one of the IP addresses already configured on the device.

Default The local loopback interface primary IPv4 address if set, else the primary IPv4 interface address of the lowest numbered VLAN the port belongs to, else the MAC address of the device's baseboard if no VLAN IP addresses are configured for the port.

Mode Interface Configuration

Usage To see the management address that will be advertised, use the show lldp interface command or show lldp local-info command.

Examples To set the management address advertised by ports 1.0.1 and 1.0.7, to be 192.168.1.6, use the commands:

```
        awplus# configure terminal
awplus(config)# interface port1.0.1,port1.0.7
awplus(config-if)# lldp management-address 192.168.1.6
```

To clear the user-configured management address advertised by ports 1.0.1 and 1.0.7, and revert to using the default address, use the commands:

```
        awplus# configure terminal
awplus(config)# interface port1.0.1,port1.0.7
awplus(config-if)# no lldp management-address
```

Related Commands

show lldp interface
show lldp local-info

lldp med-notifications

Use this command to enable LLDP to send LLDP-MED Topology Change Detected SNMP notifications relating to the specified ports. The switch sends an SNMP event notification when a new LLDP-MED compliant IP Telephony device is connected to or disconnected from a port on the switch.

Use the **no** variant of this command to disable the sending of LLDP-MED Topology Change Detected notifications relating to the specified ports.

Syntax

```
lldp med-notifications

no lldp med-notifications
```

Default The sending of LLDP-MED notifications is disabled by default.

Mode Interface Configuration

Examples To enable the sending of LLDP-MED Topology Change Detected notifications relating to ports 1.0.1 and 1.0.7, use the commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.1,port1.0.7
awplus(config-if)# lldp med-notifications
```

To disable the sending of LLDP-MED notifications relating to ports 1.0.1 and 1.0.7, use the commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.1,port1.0.7
awplus(config-if)# no lldp med-notifications
```

Related Commands

lldp notification-interval
lldp notifications
snmp-server enable trap
show lldp interface

lldp med-tlv-select

Use this command to enable LLDP-MED Organizationally Specific TLVs for transmission in LLDP advertisements via the specified ports. The LLDP-MED Capabilities TLV must be enabled before any of the other LLDP-MED Organizationally Specific TLVs are enabled.

Use the **no** variant of this command to disable the specified LLDP-MED Organizationally Specific TLVs for transmission in LLDP advertisements via these ports. In order to disable the LLDP-MED Capabilities TLV, you must also disable the rest of these TLVs. Disabling all these TLVs disables LLDP-MED advertisements.

Syntax

```
lldp med-tlv-select {[capabilities] [network-policy] [location]
    [power-management-ext] [inventory-management]}
```

```
lldp med-tlv-select all
```

```
no lldp med-tlv-select {[capabilities] [network-policy] [location]
    [power-management-ext] [inventory-management]}
```

```
no lldp med-tlv-select all
```

Parameter	Description
`capabilities`	LLDP-MED Capabilities TLV. When this is enabled, the MAC/PHY Configuration/Status TLV from IEEE 802.3 Organizationally Specific TLVs is also automatically included in LLDP-MED advertisements, whether or not it has been explicitly enabled by the lldp tlv-select command.
`network-policy`	Network Policy TLV. This TLV is transmitted if Voice VLAN parameters have been configured using the commands: ■ switchport voice dscp ■ switchport voice vlan ■ switchport voice vlan priority
`location`	Location Identification TLV. This TLV is transmitted if location information has been configured using the commands: ■ location elin-location-id ■ location civic-location identifier ■ location civic-location configuration ■ location coord-location identifier ■ location coord-location configuration ■ location elin-location
`inventory-management`	Inventory Management TLV Set, including the following TLVs: ■ Hardware Revision ■ Firmware Revision ■ Software Revision ■ Serial Number ■ Manufacturer Name ■ Model Name ■ Asset ID
`all`	All LLDP-MED Organizationally Specific TLVs.

Default By default LLDP-MED Capabilities, Network Policy, Location Identification and Extended Power-via-MDI TLVs are enabled. Therefore, if LLDP is enabled using the lldp run command, by default LLDP-MED advertisements are transmitted on ports that detect LLDP-MED neighbors connected to them.

Mode Interface Configuration

Usage LLDP-MED TLVs are only sent in advertisements via a port if there is an LLDP-MED-capable device connected to it. To see whether there are LLDP-MED capable devices connected to the ports, use the show lldp neighbors command.

Examples To enable inclusion of the Inventory TLV Set in advertisements transmitted via ports 1.0.1 and 1.0.7, use the commands:

```
         awplus# configure terminal
 awplus(config)# interface port1.0.1,port1.0.7
awplus(config-if)# lldp med-tlv-select inventory-management
```

To exclude the Inventory TLV Set in advertisements transmitted via ports 1.0.1 and 1.0.7, use the commands:

```
         awplus# configure terminal
 awplus(config)# interface port1.0.1,port1.0.7
awplus(config-if)# no lldp med-tlv-select inventory-management
```

To disable LLDP-MED advertisements transmitted via ports 1.0.1 and 1.0.7, disable all these TLVs using the commands:

```
         awplus# configure terminal
 awplus(config)# interface port1.0.1,port1.0.7
awplus(config-if)# no lldp med-tlv-select all
```

Related Commands lldp tlv-select
location elin-location-id
location civic-location identifier
location civic-location configuration
location coord-location identifier
location coord-location configuration
location elin-location
show lldp interface
switchport voice dscp
switchport voice vlan
switchport voice vlan priority

lldp non-strict-med-tlv-order-check

Use this command to enable non-strict order checking for LLDP-MED advertisements it receives. That is, use this command to enable LLDP to receive and store TLVs from LLDP-MED advertisements even if they do not use standard TLV order.

Use the **no** variant of this command to disable non-strict order checking for LLDP-MED advertisements, that is, to set strict TLV order checking, so that LLDP discards any LLDP-MED TLVs that occur before the LLDP-MED Capabilities TLV in an advertisement.

Syntax

```
lldp non-strict-med-tlv-order-check
```

```
no lldp non-strict-med-tlv-order-check
```

Default By default TLV non-strict order checking for LLDP-MED advertisements is disabled. That is, strict order checking is applied to LLDP-MED advertisements, according to ANSI/TIA-1057, and LLDP-MED TLVs in non-standard order are discarded.

Mode Global Configuration

Usage The ANSI/TIA-1057 specifies standard order for TLVs in LLDP-MED advertisements, and specifies that if LLDP receives LLDP advertisements with non-standard LLDP-MED TLV order, the TLVs in non-standard order should be discarded. This implementation of LLDP-MED follows the standard: it transmits TLVs in the standard order, and by default discards LLDP-MED TLVs that occur before the LLDP-MED Capabilities TLV in an advertisement. However, some implementations of LLDP transmit LLDP-MED advertisements with non-standard TLV order. To receive and store the data from these non-standard advertisements, enable non-strict order checking for LLDP-MED advertisements using this command.

Examples To enable strict TLV order checking, use the commands:

```
        awplus# configure terminal
awplus(config)# lldp tlv-order-check
```

To disable strict TLV order checking, use the commands:

```
        awplus# configure terminal
awplus(config)# no lldp tlv-order-check
```

Related Commands show running-config lldp

lldp notification-interval

This command sets the notification interval. This is the minimum interval between LLDP SNMP notifications (traps) of each kind (LLDP Remote Tables Change Notification and LLDP-MED Topology Change Notification).

The **no** variant of this command sets the notification interval back to its default.

Syntax

```
lldp notification-interval <5-3600>

no lldp notification-interval
```

Parameter	Description
<5-3600>	The interval in seconds.

Default The default notification interval is 5 seconds.

Mode Global Configuration

Examples To set the notification interval to 20 seconds, use the commands:

```
        awplus# configure terminal
awplus(config)# lldp notification-interval 20
```

To set the notification interval back to its default, use the commands:

```
        awplus# configure terminal
awplus(config)# no lldp notification-interval
```

Related Commands lldp notifications
show lldp

lldp notifications

This command enables the sending of LLDP SNMP notifications (traps) relating to specified ports.

The **no** variant of this command disables the sending of LLDP SNMP notifications for specified ports.

Syntax

```
lldp notifications

no lldp notifications
```

Default The sending of LLDP SNMP notifications is disabled by default.

Mode Interface Configuration

Examples To enable sending of LLDP SNMP notifications for ports 1.0.1 and 1.0.7, use the commands:

```
         awplus# configure terminal
 awplus(config)# interface port1.0.1,port1.0.7
awplus(config-if)# lldp notifications
```

To disable sending of LLDP SNMP notifications for ports 1.0.1 and 1.0.7, use the commands:

```
         awplus# configure terminal
 awplus(config)# interface port1.0.1,port1.0.7
awplus(config-if)# no lldp notifications
```

Related Commands lldp notification-interval
show lldp interface
snmp-server enable trap

lldp port-number-type

This command sets the type of port identifier used to enumerate, that is to count, the LLDP MIB local port entries. The LLDP MIB (*IEEE Standard 802.1AB-2005, Section 12, LLDP MIB Definitions.*) requires the port number value to count LLDP local port entries.

This command also enables you to optionally set an interface index to enumerate the LLDP MIB local port entries, if required by your management system.

The **no** variant of this command resets the type of port identifier back to the default setting (number).

Syntax

```
lldp port-number-type [number|ifindex]
```

```
no lldp port-number-type
```

Parameter	Description
`number`	Set the type of port identifier to a port number to enumerate the LLDP MIB local port entries.
`ifindex`	Set the type of port identifier to an interface index to enumerate the LLDP MIB local port entries.

Default The default port identifier type is number. The no variant of this command sets the port identifier type to the default.

Mode Global Configuration

Examples To set the type of port identifier used to enumerate LLDP MIB local port entries to port numbers, use the commands:

```
        awplus# configure terminal
awplus(config)# lldp port-number-type number
```

To set the type of port identifier used to enumerate LLDP MIB local port entries to interface indexes, use the commands:

```
        awplus# configure terminal
awplus(config)# lldp port-number-type ifindex
```

To reset the type of port identifier used to enumerate LLDP MIB local port entries the default (port numbers), use the commands:

```
        awplus# configure terminal
awplus(config)# no lldp port-number-type
```

Related Commands show lldp

lldp reinit

This command sets the value of the reinitialization delay. This is the minimum time after disabling LLDP on a port before it can reinitialize.

The **no** variant of this command sets the reinitialization delay back to its default setting.

Syntax

```
lldp reinit <1-10>
```

```
no lldp reinit
```

Parameter	Description
<1-10>	The delay in seconds.

Default The default reinitialization delay is 2 seconds.

Mode Global Configuration

Examples To set the reinitialization delay to 3 seconds, use the commands:

```
        awplus# configure terminal
awplus(config)# lldp reinit 3
```

To set the reinitialization delay back to its default, use the commands:

```
        awplus# configure terminal
awplus(config)# no lldp reinit
```

Related Commands show lldp

lldp run

This command enables the operation of LLDP on the device.

The **no** variant of this command disables the operation of LLDP on the device. The LLDP configuration remains unchanged.

Syntax

```
lldp run
no lldp run
```

Default

LLDP is disabled by default.

Mode

Global Configuration

Examples

To enable LLDP operation, use the commands:

```
        awplus# configure terminal
awplus(config)# lldp run
```

To disable LLDP operation, use the commands:

```
        awplus# configure terminal
awplus(config)# no lldp run
```

Related Commands

show lldp

lldp timer

This command sets the value of the transmit interval. This is the interval between regular transmissions of LLDP advertisements.

The **no** variant of this command sets the transmit interval back to its default.

Syntax

```
lldp timer <5-32768>
```

```
no lldp timer
```

Parameter	Description
<5-32768>	The transmit interval in seconds. The transmit interval must be at least four times the transmission delay timer (lldp tx-delay command).

Default The default transmit interval is 30 seconds.

Mode Global Configuration

Examples To set the transmit interval to 90 seconds, use the commands:

```
        awplus#  configure terminal
awplus(config)#  lldp timer 90
```

To set the transmit interval back to its default, use the commands:

```
        awplus#  configure terminal
awplus(config)#  no lldp timer
```

Related Commands lldp tx-delay
show lldp

lldp tlv-select

This command enables one or more optional TLVs, or all TLVs, for transmission in LLDP advertisements via the specified ports. The TLVs can be specified in any order; they are placed in LLDP frames in a fixed order (as described in IEEE 802.1AB). The mandatory TLVs (Chassis ID, Port ID, Time To Live, End of LLDPDU) are always included in LLDP advertisements.

In LLDP-MED advertisements the MAC/PHY Configuration/Status TLV will be always be included regardless of whether it is selected by this command.

The **no** variant of this command disables the specified optional TLVs, or all optional TLVs, for transmission in LLDP advertisements via the specified ports.

Syntax

```
lldp tlv-select {[<tlv>]...}

lldp tlv-select all

no lldp tlv-select {[<tlv>]...}

no lldp tlv-select all
```

Parameter	Description
`<tlv>`	The TLV to transmit in LLDP advertisements. One of these keywords: ■ port-description (specified by the description (interface) command on page 12.2) ■ system-name (specified by the hostname command on page 8.14) ■ system-description ■ system-capabilities ■ management-address ■ port-vlan ■ port-and-protocol-vlans ■ vlan-names ■ protocol-ids ■ mac-phy-config ■ power-management (Power Via MDI TLV) ■ link-aggregation ■ max-frame-size
`all`	All TLVs.

Default By default no optional TLVs are included in LLDP advertisements. The MAC/PHY Configuration/Status TLV (**mac-phy-config**) is included in LLDP-MED advertisements whether or not it is selected by this command.

Mode Interface Configuration

Examples To include the management-address and system-name TLVs in advertisements transmitted via ports 1.0.1 and 1.0.7, use the commands:

```
        awplus# configure terminal

awplus(config)# interface port1.0.1,port1.0.7

awplus(config-if)# lldp tlv-select management-address system-
                   name
```

To include all optional TLVs in advertisements transmitted via ports 1.0.1 and 1.0.7, use the commands:

```
          awplus#  configure terminal

  awplus(config)#  interface port1.0.1,port1.0.7

awplus(config-if)#  lldp tlv-select all
```

To exclude the management-address and system-name TLVs from advertisements transmitted via ports 1.0.1 and 1.0.7, use the commands:

```
          awplus#  configure terminal

  awplus(config)#  interface port1.0.1,port1.0.7

awplus(config-if)#  no lldp tlv-select management-address
                    system-name
```

To exclude all optional TLVs from advertisements transmitted via ports 1.0.1 and 1.0.7, use the commands:

```
          awplus#  configure terminal

  awplus(config)#  interface port1.0.1,port1.0.7

awplus(config-if)#  no lldp tlv-select all
```

Related Commands

description (interface)
hostname
lldp med-tlv-select
show lldp interface
show lldp local-info

lldp transmit receive

This command enables transmission and/or reception of LLDP advertisements to or from neighbors through the specified ports.

The **no** variant of this command disables transmission and/or reception of LLDP advertisements through specified ports.

Syntax

```
lldp {[transmit] [receive]}

no lldp {[transmit] [receive]}
```

Parameter	Description
`transmit`	Enable or disable transmission of LLDP advertisements via this port or ports.
`receive`	Enable or disable reception of LLDP advertisements via this port or ports.

Default LLDP advertisement transmission and reception are enabled on all ports by default.

Mode Interface Configuration

Examples To enable transmission of LLDP advertisements on ports 1.0.1 and 1.0.7, use the commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.1,port1.0.7
awplus(config-if)# lldp transmit
```

To enable LLDP advertisement transmission and reception on ports 1.0.1 and 1.0.7, use the commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.1,port1.0.7
awplus(config-if)# lldp transmit receive
```

To disable LLDP advertisement transmission and reception on ports 1.0.1 and 1.0.7, use the commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.1,port1.0.7
awplus(config-if)# no lldp transmit receive
```

Related Commands show lldp interface

lldp tx-delay

This command sets the value of the transmission delay timer. This is the minimum time interval between transmitting LLDP advertisements due to a change in LLDP local information.

The **no** variant of this command sets the transmission delay timer back to its default setting.

Syntax

```
lldp tx-delay <1-8192>
```

```
no lldp tx-delay
```

Parameter	Description
<1-8192>	The transmission delay in seconds. The transmission delay cannot be greater than a quarter of the transmit interval (lldp timer command).

Default The default transmission delay timer is 2 seconds.

Mode Global Configuration

Examples To set the transmission delay timer to 12 seconds, use the commands:

```
        awplus#  configure terminal
awplus(config)#  lldp tx-delay 12
```

To set the transmission delay timer back to its default, use the commands:

```
        awplus#  configure terminal
awplus(config)#  no lldp tx-delay
```

Related Commands lldp timer
show lldp

location civic-location configuration

Use these commands to configure a civic address location. The country parameter must be specified first, and at least one of the other parameters must be configured before the location can be assigned to a port.

Use the **no** variants of this command to delete civic address parameters from the location.

Syntax

```
country <country>

state <state>

no state

county <county>

no county

city <city>

no city

division <division>

no division

neighborhood <neighborhood>

no neighborhood

street-group <street-group>

no street-group

leading-street-direction <leading-street-direction>

no leading-street-direction

trailing-street-suffix <trailing-street-suffix>

no trailing-street-suffix

street-suffix <street-suffix>

no street-suffix

house-number <house-number>

no house-number

house-number-suffix <house-number-suffix>

no house-number-suffix

landmark <landmark>

no landmark

additional-information <additional-information>

no additional-information

name <name>

no name

postalcode <postalcode>

no postalcode
```

```
building <building>

no building

unit <unit>

no unit

floor <floor>

no floor

room <room>

no room

place-type <place-type>

no place-type

postal-community-name <postal-community-name>

no postal-community-name

post-office-box <post-office-box>

no post-office-box

additional-code <additional-code>

no additional-code

seat <seat>

no seat

primary-road-name <primary-road-name>

no primary-road-name

road-section <road-section>

no road-section

branch-road-name <branch-road-name>

no branch-road-name

sub-branch-road-name <sub-branch-road-name>

no sub-branch-road-name

street-name-pre-modifier <street-name-pre-modifier>

no street-name-pre-modifier

streetname-post-modifier <streetname-post-modifier>

no streetname-post-modifier
```

Parameter	Description
<country>	Upper-case two-letter country code, as specified in *ISO 3166*.
<state>	State (Civic Address (CA) Type 1): national subdivisions (state, canton, region).
<county>	County (CA Type 2): County, parish, gun (JP), district (IN).

Parameter(cont.)	**Description(cont.)**
<city>	City (CA Type 3): city, township, shi (JP).
<division>	City division (CA Type 4): City division, borough, city district, ward, chou (JP).
<neighborhood>	Neighborhood (CA Type 5): neighborhood, block.
<street-group>	Street group (CA Type 6): group of streets below the neighborhood level.
<leading-street-direction>	Leading street direction (CA Type 16).
<trailing-street-suffix>	Trailing street suffix (CA Type 17).
<street-suffix>	Street suffix (CA Type 18): street suffix or type.
<house-number>	House number (CA Type 19).
<house-number-suffix>	House number suffix (CA Type 20).
<landmark>	Landmark or vanity address (CA Type 21).
<additional-information>	Additional location information (CA Type 22).
<name>	Name (CA Type 23): residence and office occupant.
<postal-code>	Postal/zip code (CA Type 24).
<building>	Building (CA Type 25): structure.
<unit>	Unit (CA Type 26): apartment, suite.
<floor>	Floor (CA Type 27).
<room>	Room (CA Type 28).
<place-type>	Type of place (CA Type 29).
<postal-community-name>	Postal community name (CA Type 30).
<post-office-box>	Post office box (P.O. Box) (CA Type 31).
<additional-code>	Additional code (CA Type 32).
<seat>	Seat (CA Type 33): seat (desk, cubicle, workstation).
<primary-road-name>	Primary road name (CA Type 34).
<road-section>	Road section (CA Type 35).
<branch-road-name>	Branch road name (CA Type 36).
<sub-branch-road-name>	Sub-branch road name (CA Type 37).
<street-name-pre-modifier>	Street name pre-modifier (CA Type 38).
<street-name-post-modifier>	Street name post-modifier (CA Type 39).

Default By default no civic address location information is configured.

Mode Civic Address Location Configuration

Usage The **country** parameter must be configured before any other parameters can be configured; this creates the location. The country parameter cannot be deleted. One or more of the other parameters must be configured before the location can be assigned to a port. The country parameter must be entered as an upper-case two-letter country code, as specified in *ISO 3166*. All other parameters are entered as alpha-numeric strings. Do not configure all the civic address parameters (this would generate TLVs that are too long). Configure a subset of these parameters—enough to consistently and precisely identify the location of the device. If the location is to be used for Emergency Call Service (ECS), the particular ECS application may have guidelines for configuring the civic address location. For more information about civic address format, see "LLDP-MED: Location Identification TLV" on page 65.7.

To specify the civic address location, use the location civic-location identifier command. To delete the civic address location, use the **no** variant of the **location civic-location identifier** command. To assign the civic address location to particular ports, so that it can be advertised in TLVs from those ports, use the command location civic-location-id command.

Examples To configure civic address location 1 with location "27 Nazareth Avenue, Christchurch, New Zealand" in civic-address format, use the commands:

```
               awplus# configure terminal
       awplus(config)# location civic-location identifier 1
 awplus(config-civic)# country NZ
 awplus(config-civic)# city Christchurch
 awplus(config-civic)# primary-road-name Nazareth
 awplus(config-civic)# street-suffix Avenue
 awplus(config-civic)# house-number 27
```

Related Commands location civic-location-id
location civic-location identifier
show lldp local-info
show location

location civic-location identifier

Use this command to enter the Civic Address Location Configuration mode to configure the specified location.

Use the **no** variant of this command to delete a civic address location. This also removes the location from any ports it has been assigned to.

Syntax

```
location civic-location identifier <civic-loc-id>
no location civic-location identifier <civic-loc-id>
```

Parameter	Description
<civic-loc-id>	A unique civic address location ID, in the range 1 to 4095.

Default By default there are no civic address locations.

Mode Global Configuration

Usage To configure the location information for this civic address location identifier, use the location civic-location configuration command. To associate this civic location identifier with particular ports, use the location elin-location-id command.

Up to 400 locations can be configured on the switch for each type of location information, up to a total of 1200 locations.

Examples To enter Civic Address Location Configuration mode for the civic address location with ID 1, use the commands:

```
           awplus# configure terminal
   awplus(config)# location civic-location identifier 1
awplus(config-civic)#
```

To delete the civic address location with ID 1, use the commands:

```
         awplus# configure terminal
 awplus(config)# no location civic-location identifier 1
```

Related Commands

location civic-location-id
location civic-location configuration
show location
show running-config lldp

location civic-location-id

Use this command to assign a civic address location to the ports. The civic address location must already exist. This replaces any previous assignment of civic address location for the ports. Up to one location of each type can be assigned to a port.

Use the **no** variant of this command to remove a location identifier from the ports.

Syntax

```
location civic-location-id <civic-loc-id>

no location civic-location-id [<civic-loc-id>]
```

Parameter	Description
<civic-loc-id>	Civic address location ID, in the range 1 to 4095.

Default By default no civic address location is assigned to ports.

Mode Interface Configuration

Usage The civic address location associated with a port can be transmitted in Location Identification TLVs via the port.

Before using this command, create the location using the following commands:

- location civic-location identifier command
- location civic-location configuration command

If a civic-address location is deleted using the **no** variant of the location civic-location identifier command, it is automatically removed from all ports.

Examples To assign the civic address location 1 to port1.0.1, use the commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.1
awplus(config-if)# location civic-location-id 1
```

To remove a civic address location from port1.0.1, use the commands:

```
          awplus# configure terminal
  awplus(config)# interface port1.0.1
awplus(config-if)# no location civic-location-id
```

Related Commands lldp med-tlv-select
location civic-location identifier
location civic-location configuration
show location

location coord-location configuration

Use this command to configure a coordinate-based location. All parameters must be configured before assigning this location identifier to a port.

Syntax

```
latitude <latitude>

lat-resolution <lat-resolution>

longitude <longitude>

long-resolution <long-resolution>

altitude <altitude> {meters|floor}

alt-resolution <alt-resolution>

datum {wgs84|nad83-navd|nad83-mllw}
```

Parameter	Description
<lat-resolution>	Latitude resolution, as a number of valid bits, in the range 0 to 34.
<latitude>	Latitude value in degrees in the range -90.0 to 90.0
<long-resolution>	Longitude resolution, as a number of valid bits, in the range 0 to 34.
<longitude>	Longitude value in degrees, in the range -180.0 to 180.0.
<alt-resolution>	Altitude resolution, as a number of valid bits, in the range 0 to 30. A resolution of 0 can be used to indicate an unknown value.
<altitude>	Altitude value, in meters or floors.
`meters`	The altitude value is in meters.
`floors`	The altitude value is in floors.
`datum`	The geodetic system (or datum) that the specified coordinate values are based on.
`wgs84`	World Geodetic System 1984.
`nad83-navd`	North American Datum 1983 - North American Vertical Datum.
`nad83-mllw`	North American Datum 1983 - Mean Lower Low Water vertical datum.

Default By default no coordinate location information is configured.

Mode Coordinate Configuration

Usage Latitude and longitude values are always stored internally, and advertised in the Location Identification TLV, as 34-bit fixed-point binary numbers, with a 25-bit fractional part, irrespective of the number of digits entered by the user. Likewise altitude is stored as a 30-bit fixed point binary number, with an 8-bit fractional part. Because the user-entered decimal values are stored as fixed point binary numbers, they cannot always be represented exactly—the stored binary number is converted to a decimal number for display in the output of the show location command. For example, a user-entered latitude value of "2.77" degrees is displayed as "2.76999998092651367187500000".

The **lat-resolution**, **long-resolution**, and **alt-resolution** parameters allow the user to specify the resolution of each coordinate element as the number of valid bits in the internally-stored binary representation of the value. These resolution values can be used by emergency services to define a search area.

To specify the coordinate identifier, use the location coord-location identifier command. To remove coordinate information, delete the coordinate location by using the **no** variant of that command. To associate the coordinate location with particular ports, so that it can be advertised in TLVs from those ports, use the location elin-location-id command.

Example To configure the location for the White House in Washington DC, which has the coordinates based on the WGS84 datum of 38.89868 degrees North (with 22 bit resolution), 77.03723 degrees West (with 22 bit resolution), and 15 meters height (with 9 bit resolution), use the commands:

```
               awplus# configure terminal
       awplus(config)# location coord-location identifier 1
 awplus(config-coord)# la-resolution 22
 awplus(config-coord)# latitude 38.89868
 awplus(config-coord)# lo-resolution 22
 awplus(config-coord)# longitude -77.03723
 awplus(config-coord)# alt-resolution 9
 awplus(config-coord)# altitude 15 meters
 awplus(config-coord)# datum wgs84
```

Related Commands location coord-location-id
location coord-location identifier
show lldp local-info
show location

location coord-location identifier

Use this command to enter Coordinate Location Configuration mode for this coordinate location.

Use the **no** variant of this command to delete a coordinate location. This also removes the location from any ports it has been assigned to.

Syntax

```
location coord-location identifier <coord-loc-id>

no location coord-location identifier <coord-loc-id>
```

Parameter	Description
<coord-loc-id>	A unique coordinate location identifier, in the range 1 to 4095.

Default By default there are no coordinate locations.

Mode Global Configuration

Usage Up to 400 locations can be configured on the switch for each type of location information, up to a total of 1200 locations.

To configure this coordinate location, use the location coord-location configuration command. To associate this coordinate location with particular ports, so that it can be advertised in TLVs from those ports, use the location coord-location-id command.

Examples To enter Coordinate Location Configuration mode to configure the coordinate location with ID 1, use the commands:

```
         awplus# configure terminal
  awplus(config)# location coord-location identifier 1
awplus(config-coord)#
```

To delete coordinate location 1, use the commands:

```
         awplus# configure terminal
  awplus(config)# no location coord-location identifier 1
```

Related Commands

location coord-location-id
location coord-location configuration
show lldp local-info
show location

location coord-location-id

Use this command to assign a coordinate location to the ports. The coordinate location must already exist. This replaces any previous assignment of coordinate location for the ports. Up to one location of each type can be assigned to a port.

Use the **no** variant of this command to remove a location from the ports.

Syntax

```
location coord-location-id <coord-loc-id>
```

```
no location coord-location-id [<coord-loc-id>]
```

Parameter	Description
<coord-loc-id>	Coordinate location ID, in the range 1 to 4095.

Default By default no coordinate location is assigned to ports.

Mode Interface Configuration

Usage The coordinate location associated with a port can be transmitted in Location Identification TLVs via the port.

Before using this command, configure the location using the following commands:

- location coord-location identifier command
- location coord-location configuration command

If a coordinate location is deleted using the **no** variant of the location coord-location identifier command, it is automatically removed from all ports.

Examples To assign coordinate location 1 to port1.0.1, use the commands:

```
          awplus#  configure terminal
  awplus(config)#  interface port1.0.1
awplus(config-if)#  location coord-location-id 1
```

To remove a coordinate location from port1.0.1, use the commands:

```
          awplus#  configure terminal
  awplus(config)#  interface port1.0.1
awplus(config-if)#  no location coord-location-id
```

Related Commands lldp med-tlv-select
location coord-location identifier
location coord-location configuration
show location

location elin-location

Use this command to create or modify an ELIN location.

Use the **no** variant of this command to delete an ELIN location, and remove it from any ports it has been assigned to.

Syntax

```
location elin-location <elin> identifier <elin-loc-id>

no location elin-location identifier <elin-loc-id>
```

Parameter	Description
<elin>	Emergency Location Identification Number (ELIN) for Emergency Call Service (ECS), in the range 10 to 25 digits long. In North America, ELINs are typically 10 digits long.
<elin-loc-id>	A unique ELIN location identifier, in the range 1 to 4095.

Default By default there are no ELIN location identifiers.

Mode Global Configuration

Usage Up to 400 locations can be configured on the switch for each type of location information, up to a total of 1200 locations.

To assign this ELIN location to particular ports, so that it can be advertised in TLVs from those ports, use the location elin-location-id command.

Examples To create a new ELIN location with ID 1, and configure it with ELIN "1234567890", use the commands:

```
        awplus# configure terminal
awplus(config)# location elin-location 1234567890 identifier 1
```

To delete existing ELIN location with ID 1, use the commands:

```
        awplus# configure terminal
awplus(config)# no location elin-location identifier 1
```

Related Commands

location elin-location-id
show lldp local-info
show location

location elin-location-id

Use this command to assign an ELIN location to the ports. The ELIN location must already exist. This replaces any previous assignment of ELIN location for the ports. Up to one location of each type can be assigned to a port.

Use the **no** variant of this command to remove a location identifier from the ports.

Syntax

```
location elin-location-id <elin-loc-id>

no location elin-location-id [<elin-loc-id>]
```

Parameter	Description
<elin-loc-id>	ELIN location identifier, in the range 1 to 4095.

Default By default no ELIN location is assigned to ports.

Mode Interface Configuration

Usage An ELIN location associated with a port can be transmitted in Location Identification TLVs via the port.

Before using this command, configure the location using the location elin-location command.

If an ELIN location is deleted using the **no** variant of one of the location elin-location command, it is automatically removed from all ports.

Examples To assign ELIN location 1 to port 1.0.1, use the commands:

```
        awplus#  configure terminal
awplus(config)#  interface port1.0.1
awplus(config-if)#  location elin-location-id 1
```

To remove an ELIN location from port 1.0.1, use the commands:

```
        awplus#  configure terminal
awplus(config)#  interface port1.0.1
awplus(config-if)#  no location elin-location-id
```

Related Commands lldp med-tlv-select
location elin-location
show location

show debugging lldp

This command displays LLDP debug settings for specified ports. If no port list is supplied, LLDP debug settings for all ports are displayed.

Syntax `show debugging lldp [interface <port-list>]`

Parameter	Description
<port-list>	The ports for which the LLDP debug settings are shown.

Mode User Exec and Privileged Exec

Examples To display LLDP debug settings for all ports, use the command:

```
awplus# show debugging lldp
```

To display LLDP debug settings for ports 1.0.1 to 1.0.9, use the command:

```
awplus# show debugging lldp interface port1.0.1-1.0.9
```

Output Figure 66-1: Example output from the **show debugging lldp** command

```
LLDP Debug settings:
Debugging for LLDP internal operation is on
Port       Rx     RxPkt    Tx     TxPkt
------------------------------------
1.0.1      Yes    Yes      No     No
1.0.2      Yes    No       No     No
1.0.3      No     No       No     No
1.0.4      Yes    Yes      Yes    No
1.0.5      Yes    No       Yes    No
1.0.6      No     No       Yes    No
1.0.7      Yes    Yes      Yes    Yes
1.0.8      Yes    No       Yes    Yes
1.0.9      No     No       Yes    Yes
```

Table 66-1: Parameters in the output of the **show debugging lldp** command

Parameter	Description
`Port`	Port name.
`Rx`	Whether debugging of LLDP receive is enabled on the port.
`RxPkt`	Whether debugging of LLDP receive packet dump is enabled on the port.
`Rx`	Whether debugging of LLDP transmit is enabled on the port.
`RxPkt`	Whether debugging of LLDP transmit packet dump is enabled on the port.

Related Commands debug lldp

show lldp

This command displays LLDP status and global configuration settings.

Syntax `show lldp`

Mode User Exec and Privileged Exec

Example To display LLDP status and global configuration settings, use the command:

```
awplus# show lldp
```

Output

Figure 66-2: Example output from the **show lldp** command

```
awplus# show lldp

LLDP Global Configuration:                [Default Values]
LLDP Status ............... Enabled       [Disabled]
Notification Interval ..... 5 secs        [5]
Tx Timer Interval ......... 30 secs       [30]
Hold-time Multiplier ...... 4             [4]
(Computed TTL value ....... 120 secs)
Reinitialization Delay .... 2 secs        [2]
Tx Delay .................. 2 secs        [2]
Port Number Type........... Ifindex       [Port-Number]
Fast Start Count .......... 5             [3]

LLDP Global Status:
Total Neighbor Count ...... 47
Neighbors table last updated 0 hrs 0 mins 43 secs ago
```

Table 66-2: Parameters in the output of the **show lldp** command

Parameter	Description
`LLDP Status`	Whether LLDP is enabled. Default is disabled.
`Notification Interval`	Minimum interval between LLDP notifications.
`Tx Timer Interval`	Transmit interval between regular transmissions of LLDP advertisements.
`Hold-time Multiplier`	The holdtime multiplier. The transmit interval is multiplied by the holdtime multiplier to give the Time To Live (TTL) value that is advertised to neighbors.
`Reinitialization Delay`	The reinitialization delay. This is the minimum time after disabling LLDP transmit on a port before it can reinitialize again.
`Tx Delay`	The transmission delay. This is the minimum time interval between transmitting advertisements due to a change in LLDP local information.
`Port Number Type`	The type of port identifier used to enumerate LLDP MIB local port entries, as set by the lldp port-number-type command.
`Fast Start Count`	The number of times fast start advertisements are sent for LLDP-MED.

Table 66-2: Parameters in the output of the **show lldp** command(cont.)

Parameter	Description
`Total Neighbor Count`	Number of LLDP neighbors discovered on all ports.
`Neighbors table last updated`	The time since the LLDP neighbor table was last updated.

Related Commands show lldp interface
show running-config lldp

show lldp interface

This command displays LLDP configuration settings for specified ports. If no port list is specified, LLDP configuration for all ports is displayed.

Syntax `show lldp interface [<port-list>]`

Parameter	Description
<port-list>	The ports for which the LLDP configuration settings are to be shown.

Mode User Exec and Privileged Exec

Examples To display LLDP configuration settings for ports 1.0.1 to 1.0.8, use the command:

```
awplus# show lldp interface port1.0.1-1.0.8
```

To display LLDP configuration settings for all ports, use the command:

```
awplus# show lldp interface
```

Output

Figure 66-3: Example output from the **show lldp interface** command

```
awplus# show lldp interface port1.0.1-1.0.8
LLDP Port Status and Configuration:

  * = LLDP is inactive on this port because it is a mirror analyser port
  Notification Abbreviations:
    RC = LLDP Remote Tables Change       TC = LLDP-MED Topology Change
  TLV Abbreviations:
    Base:  Pd = Port Description         Sn = System Name
           Sd = System Description       Sc = System Capabilities
           Ma = Management Address
    802.1: Pv = Port VLAN ID             Pp = Port And Protocol VLAN ID
           Vn = VLAN Name                Pi = Protocol Identity
    802.3: Mp = MAC/PHY Config/Status    Po = Power Via MDI (PoE)
           La = Link Aggregation         Mf = Maximum Frame Size
    MED:   Mc = LLDP-MED Capabilities    Np = Network Policy
           Lo = Location Identification  Pe = Extended PoE    In = Inventory

                                           Optional TLVs Enabled for Tx
 Port     Rx/Tx  Notif  Management Addr  Base       802.1    802.3    MED
-------------------------------------------------------------------------------
 1.0.1    Rx Tx  RC --  192.168.100.123  PdSnSdScMa -------- -------- McNpLoPe--
*1.0.2    -- Tx  RC --  192.168.100.123  PdSnSdScMa -------- -------- McNpLoPe--
 1.0.3    Rx Tx  RC --  192.168.100.123  Pd--SdScMa PvPpVnPi -------- McNpLoPe--
 1.0.4    -- --  RC --  192.168.100.123  PdSnSd--Ma -------- -------- McNpLoPe--
 1.0.5    Rx Tx  RC TC  192.168.100.123  PdSnSdScMa PvPpVnPi -------- McNpLoPe--
 1.0.6    Rx Tx  RC TC  192.168.100.123  Pd----ScMa -------- -------- McNpLoPe--
 1.0.7    Rx Tx  -- TC  192.168.100.123  PdSnSdScMa PvPpVnPi MpPoLaMf McNpLoPeIn
 1.0.8    Rx Tx  -- TC  192.168.1.1      PdSn--ScMa PvPpVnPi -------- McNp------
```

Table 66-3: Parameters in the output of the **show lldp interface** command

Parameter	Description
`Port`	Port name.
`Rx`	Whether reception of LLDP advertisements is enabled on the port.
`Tx`	Whether transmission of LLDP advertisements is enabled on the port.
`Notif`	Whether sending SNMP notification for LLDP is enabled on the port: ■ RM = Remote Tables Change Notification ■ TP = LLDP-MED Topology Change Notification
`Management Addr`	Management address advertised to neighbors.
`Base TLVs Enabled for Tx`	List of optional Base TLVs enabled for transmission: ■ Pd = Port Description ■ Sn =System Name ■ Sd = System Description ■ Sc =System Capabilities ■ Ma = Management Address
`802.1 TLVs Enabled for Tx`	List of optional 802.1 TLVs enabled for transmission: ■ Pv = Port VLAN ID ■ Pp = Port And Protocol VLAN ID ■ Vn = VLAN Name ■ Pi =Protocol Identity
`802.3 TLVs Enabled for Tx`	List of optional 802.3 TLVs enabled for transmission: ■ Mp = MAC/PHY Configuration/Status ■ Po = Power Via MDI (PoE) ■ La = Link Aggregation ■ Mf = Maximum Frame Size
`MED TLVs Enabled for Tx`	List of optional LLDP-MED TLVs enabled for transmission: ■ Mc = LLDP-MED Capabilities ■ Np = Network Policy ■ Lo = Location Information, ■ Pe = Extended Power-Via-MDI ■ In = Inventory

Related Commands

show lldp
show running-config lldp

show lldp local-info

This command displays local LLDP information that can be transmitted through specified ports. If no port list is entered, local LLDP information for all ports is displayed.

Syntax

```
show lldp local-info [base] [dot1] [dot3] [med] [interface
    <port-list>]
```

Parameter	Description
`base`	Information for base TLVs.
`dot1`	Information for 802.1 TLVs.
`dot3`	Information for 802.3 TLVs.
`med`	Information for LLDP-MED TLVs.
`<port-list>`	The ports for which the local information is to be shown.

Mode User Exec and Privileged Exec

Usage Whether and which local information is transmitted in advertisements via a port depends on:

- whether the port is set to transmit LLDP advertisements (lldp transmit receive command)
- which TLVs it is configured to send (lldp tlv-select command, lldp med-tlv-select command)

Examples To display local information transmitted via port 1.0.1, use the command:

```
awplus# show lldp local-info interface port1.0.1
```

To display local information transmitted via all ports, use the command:

```
awplus# show lldp local-info
```

Output

Figure 66-4: Example output from the **show lldp local-info** command

```
LLDP Local Information:

Local port1.0.1:
  Chassis ID Type .................. MAC address
  Chassis ID ....................... 0015.77c9.7453
  Port ID Type ..................... Interface alias
  Port ID .......................... port1.0.1
  TTL .............................. 120
  Port Description ................. [not configured]
  System Name ...................... awplus
  System Description ............... Allied Telesis router/switch, AW+
                                     v5.3.3
  System Capabilities - Supported .. Bridge, Router
                      - Enabled .... Bridge, Router
  Management Address ............... 192.168.1.6
  Port VLAN ID (PVID) .............. 1
  Port & Protocol VLAN - Supported . Yes
                       - Enabled ... No
                       - VIDs ...... 0
  VLAN Names ....................... default
  Protocol IDs ..................... 9000, 0026424203000000, 888e01, aaaa03,
                                     88090101, 00540000e302, 0800, 0806, 86dd
  MAC/PHY Auto-negotiation ......... Supported, Enabled
       Advertised Capability ....... 1000BaseTFD, 100BaseTXFD, 100BaseTX,
                                     10BaseTFD, 10BaseT
       Operational MAU Type ........ 1000BaseTFD (30)
  Power Via MDI (PoE) .............. Supported, Enabled
       Port Class .................. PSE
       Pair Control Ability ........ Disabled
       Power Class ................. Unknown
  Link Aggregation ................. Supported, Disabled
  Maximum Frame Size ............... 1522
  LLDP-MED Device Type ............. Network Connectivity
  LLDP-MED Capabilities ............ LLDP-MED Capabilities, Network Policy,
                                     Location Identification,
                                     Extended Power - PSE, Inventory
  Network Policy ................... [not configured]
  Location Identification .......... Civic Address
       Country Code ................ NZ
       City ........................ Christchurch
       Street Suffix ............... Avenue
       House Number ................ 27
       Primary Road Name ........... Nazareth
  Location Identification .......... ELIN
       ELIN ........................ 123456789012
  Extended Power Via MDI (PoE) ..... PSE
       Power Source ................ Primary Power
       Power Priority .............. Low
       Power Value ................. 4.4 Watts
  Inventory Management:
       Hardware Revision ........... A-0
       Firmware Revision ........... 1.1.0
       Software Revision ........... v5.3.3
       Serial Number ............... G1Q78900B
       Manufacturer Name ........... Allied Telesis Inc.
       Model Name .................. x600-48Ts/XP
       Asset ID .................... [zero length]
```

Table 66-4: Parameters in the output of the **show lldp local-info** command

Parameter	Description
`Chassis ID Type`	Type of the Chassis ID.
`Chassis ID`	Chassis ID that uniquely identifies the local device.
`Port ID Type`	Type of the Port ID.
`Port ID`	Port ID of the local port through which advertisements are sent.
`TTL`	Number of seconds that the information advertised by the local port remains valid.
`Port Description`	Port description of the local port, as specified by the **description (interface)** command on page 12.2.
`System Name`	System name, as specified by the **hostname** command on page 8.14.
`System Description`	System description.
`System Capabilities (Supported)`	Capabilities that the local port supports.
`System Capabilities (Enabled)`	Enabled capabilities on the local port.
`Management Addresses`	Management address associated with the local port. To change this, use the **lldp management-address** command.
`Port VLAN ID (PVID)`	VLAN identifier associated with untagged or priority tagged frames received via the local port.
`Port & Protocol VLAN (Supported)`	Whether Port & Protocol VLANs (PPV) is supported on the local port.
`Port & Protocol VLAN (Enabled)`	Whether the port is in one or more Port & Protocol VLANs.
`Port & Protocol VLAN (VIDs)`	List of identifiers for Port & Protocol VLANs that the port is in.
`VLAN Names`	List of VLAN names for VLANs that the local port is assigned to.
`Protocol IDs`	List of protocols that are accessible through the local port.
`MAC/PHY Auto-negotiation`	Auto-negotiation support and current status of the 802.3 LAN on the local port.
`Power Via MDI (PoE)`	PoE-capability and current status on the local port.
`Port Class`	Whether the device is a PSE (Power Sourcing Entity) or a PD (Powered Device)
`Pair Control Ability`	Whether power pair selection can be controlled
`Power Pairs`	Which power pairs are selected for power ("Signal Pairs" or "Spare Pairs") if pair selection can be controlled

Table 66-4: Parameters in the output of the **show lldp local-info** command(cont.)

Parameter	Description
`Power Class`	The power class of the PD device on the port (class 0, 1, 2, 3 or 4)
`Link Aggregation`	Whether the link is capable of being aggregated and it is currently in an aggregation.
`Aggregated Port-ID`	Aggregated port identifier.
`Maximum Frame Size`	The maximum frame size capability of the implemented MAC and PHY.
`LLDP-MED Device Type`	LLDP-MED device type
`LLDP-MED Capabilities`	Capabilities LLDP-MED capabilities supported on the local port.
`Network Policy`	List of network policies configured on the local port.
`VLAN ID`	VLAN identifier for the port for the specified application type
`Tagged Flag`	Whether the VLAN ID is to be used as tagged or untagged
`Layer-2 Priority:`	Layer 2 User Priority (in the range 0 to 7)
`DSCP Value`	Diffserv codepoint (in the range 0 to 63)
`Location Identification`	Location configured on the local port.
`Extended Power Via MDI (PoE)`	PoE-capability and current status of the PoE parameters for Extended Power-Via-MDI TLV on the local port.
`Power Source`	The power source the switch currently uses; either primary power or backup power.
`Power Priority`	The power priority configured on the port; either critical, high or low.
`Power Value`	The total power the switch can source over a maximum length cable to a PD device on the port. The value shows the power value in Watts from the PD side.
`Inventory Management`	Inventory information for the device.

Related Commands

description (interface)
hostname
lldp transmit receive

show lldp neighbors

This command displays a summary of information received from neighbors via specified ports. If no port list is supplied, neighbor information for all ports is displayed.

Syntax

```
show lldp neighbors [interface <port-list>]
```

Parameter	Description
<port-list>	The ports for which the neighbor information is to be shown.

Mode User Exec and Privileged Exec

Examples To display neighbor information received via all ports, use the command:

```
awplus# show lldp neighbors
```

To display neighbor information received via ports 1.0.1 and 1.0.7 with LLDP-MED configuration, use the command:

```
awplus# show lldp neighbors interface port1.0.1,port1.0.7
```

Output

Figure 66-5: Example output from the **show lldp neighbors** command

```
LLDP Neighbor Information:

Total number of neighbors on these ports .... 4

  System Capability Codes:
    O = Other   P = Repeater   B = Bridge                W = WLAN Access Point
    R = Router  T = Telephone  C = DOCSIS Cable Device   S = Station Only
  LLDP-MED Device Type and Power Source Codes:
    1 = Class I   3 = Class III    PSE  = PoE    Both = PoE&Local   Prim = Primary
    2 = Class II  N = Network Con. Locl = Local  Unkn = Unknown     Back = Backup

Local   Neighbor        Neighbor        Neighbor                System      MED
Port    Chassis ID      Port ID         Sys Name                Cap.        Ty Pwr
----------------------------------------------------------------------------------
1.0.1   002d.3044.7ba6  port1.0.2       awplus                  OPBWRTCS
1.0.1   0011.3109.e5c6  port1.0.3       AT-9924 switch/route... --B-R---
1.0.7   0000.10cf.8590  port3           AR-442S                 --B-R---
1.0.7   00ee.4352.df51  192.168.1.2     Jim's desk phone        --B--T--    3  PSE
```

Table 66-5: Parameters in the output of the **show lldp neighbors** command

Parameter	Description
`Local Port`	Local port on which the neighbor information was received.
`Neighbor Chassis ID`	Chassis ID that uniquely identifies the neighbor.
`Neighbor Port Name`	Port ID of the neighbor.
`Neighbor Sys Name`	System name of the LLDP neighbor.

Table 66-5: Parameters in the output of the **show lldp neighbors** command(cont.)

Parameter	Description
`Neighbor Capability`	Capabilities that are supported and enabled on the neighbor.
`System Capability`	System Capabilities of the LLDP neighbor.
`MED Device Type`	LLDP-MED Device class (Class I, II, III or Network Connectivity)
`MED Power Source`	LLDP-MED Power Source

Related Commands show lldp neighbors detail

show lldp neighbors detail

This command displays in detail the information received from neighbors via specified ports. If no port list is supplied, detailed neighbor information for all ports is displayed.

Syntax

```
show lldp neighbors detail [base] [dot1] [dot3] [med] [interface
    <port-list>]
```

Parameter	Description
`base`	Information for base TLVs.
`dot1`	Information for 802.1 TLVs.
`dot3`	Information for 803.1 TLVs.
`med`	Information for LLDP-MED TLVs.
`<port-list>`	The ports for which the neighbor information is to be shown.

Mode User Exec and Privileged Exec

Examples To display detailed neighbor information received via all ports, use the command:

```
awplus# show lldp neighbors detail
```

To display detailed neighbor information received via ports 1.0.1, use the command:

```
awplus# show lldp neighbors detail interface port1.0.1
```

Output

Figure 66-6: Example output from the **show lldp neighbors detail** command

```
awplus# show lldp neighbors detail interface port1.0.1
LLDP Detailed Neighbor Information:

Local port1.0.1:
  Neighbors table last updated 0 hrs 0 mins 40 secs ago

  Chassis ID Type .................. MAC address
  Chassis ID ....................... 0004.cd28.8754
  Port ID Type ..................... Interface alias
  Port ID .......................... port1.0.8
  TTL .............................. 120 (secs)
  Port Description ................. [zero length]
  System Name ...................... awplus
  System Description ............... Allied Telesis router/switch, AW+ v5.3.3
  System Capabilities - Supported .. Bridge, Router
                      - Enabled .... Bridge, Router
  Management Addresses ............ 0004.cd28.8754
  Port VLAN ID (PVID) .............. 1
  Port & Protocol VLAN - Supported . Yes
                       - Enabled ... Yes
                       - VIDs ...... 5
  VLAN Names ....................... default, vlan5
  Protocol IDs ..................... 9000, 0026424203000000, 888e01, 8100,
                                     88090101, 00540000e302, 0800, 0806, 86dd
  MAC/PHY Auto-negotiation ......... Supported, Enabled
       Advertised Capability ....... 1000BaseTFD, 100BaseTXFD, 100BaseTX,
                                     10BaseTFD, 10BaseT
       Operational MAU Type ........ 1000BaseTFD (30)
  Power Via MDI (PoE) .............. [not advertised]
  Link Aggregation ................. Supported, Disabled
  Maximum Frame Size ............... 1522 (Octets)
  LLDP-MED Device Type ............. Network Connectivity
  LLDP-MED Capabilities ............ LLDP-MED Capabilities, Network Policy,
                                     Location Identification,
                                     Extended Power - PSE, Inventory
  Network Policy ................... [not advertised]
  Location Identification .......... [not advertised]
  Extended Power Via MDI (PoE) ..... PD
           Power Source ............ PSE
           Power Priority .......... High
           Power Value ............. 4.4 Watts
  Inventory Management:
           Hardware Revision ....... X1-0
           Firmware Revision ....... 1.1.0
           Software Revision ....... 5.3.3
           Serial Number ........... M1NB73008
           Manufacturer Name ....... Allied Telesis Inc.
           Model Name .............. x900-12XT/S
           Asset ID ................ [zero length]
```

Table 66-6: Parameters in the output of the **show lldp neighbors detail** command

Parameter	Description
`Chassis ID Type`	Type of the Chassis ID.
`Chassis ID`	Chassis ID that uniquely identifies the neighbor.
`Port ID Type`	Type of the Port ID.
`Port ID`	Port ID of the neighbor.
`TTL`	Number of seconds that the information advertised by the neighbor remains valid.
`Port Description`	Port description of the neighbor's port.
`System Name`	Neighbor's system name.

Table 66-6: Parameters in the output of the **show lldp neighbors detail** command(cont.)

Parameter	Description
`System Description`	Neighbor's system description.
`System Capabilities (Supported)`	Capabilities that the neighbor supports.
`System Capabilities (Enabled)`	Capabilities that are enabled on the neighbor.
`Management Addresses`	List of neighbor's management addresses.
`Port VLAN ID (PVID)`	VLAN identifier associated with untagged or priority tagged frames for the neighbor port.
`Port & Protocol VLAN (Supported)`	Whether Port & Protocol VLAN is supported on the LLDP neighbor.
`Port & Protocol VLAN (Enabled)`	Whether Port & Protocol VLAN is enabled on the LLDP neighbor.
`Port & Protocol VLAN (VIDs)`	List of Port & Protocol VLAN identifiers.
`VLAN Names`	List of names of VLANs that the neighbor's port belongs to.
`Protocol IDs`	List of protocols that are accessible through the neighbor's port.
`MAC/PHY Auto-negotiation`	Auto-negotiation configuration and status
`Power Via MDI (PoE)`	PoE configuration and status of 802.3 Power-Via-MDI TLV
`Link Aggregation`	Link aggregation information
`Maximum Frame Size`	The maximum frame size capability
`LLDP-MED Device Type`	LLDP-MED Device type
`LLDP-MED Capabilities`	LLDP-MED capabilities supported
`Network Policy`	List of network policies
`Location Identification`	Location information
`Extended Power Via MDI (PoE)`	PoE-capability and current status
`Inventory Management`	Inventory information

Related Commands show lldp neighbors

show lldp statistics

This command displays the global LLDP statistics (packet and event counters).

Syntax

```
show lldp statistics
```

Mode User Exec and Privileged Exec

Example To display global LLDP statistics information, use the command:

```
awplus# show lldp statistics
```

Output

Figure 66-7: Example output from the **show lldp statistics** command

```
awplus# show lldp statistics

Global LLDP Packet and Event counters:

  Frames:    Out ................... 345
             In .................... 423
             In Errored ............ 0
             In Dropped ............ 0
  TLVs:      Unrecognized .......... 0
             Discarded ............. 0
  Neighbors: New Entries ........... 20
             Deleted Entries ....... 20
             Dropped Entries ....... 0
             Entry Age-outs ........ 20
```

Table 66-7: Parameters in the output of the **show lldp statistics** command

Parameter	Description
`Frames Out`	Number of LLDPDU frames transmitted.
`Frames In`	Number of LLDPDU frames received.
`Frames In Errored`	Number of invalid LLDPDU frames received.
`Frames In Dropped`	Number of LLDPDU frames received and discarded for any reason.
`TLVs Unrecognized`	Number of LLDP TLVs received that are not recognized but the TLV type is in the range of reserved TLV types.
`TLVs Discarded`	Number of LLDP TLVs discarded for any reason.
`Neighbors New Entries`	Number of times the information advertised by neighbors has been inserted into the neighbor table.
`Neighbors Deleted Entries`	Number of times the information advertised by neighbors has been removed from the neighbor table.
`Neighbors Dropped Entries`	Number of times the information advertised by neighbors could not be entered into the neighbor table because of insufficient resources.
`Neighbors Entry Age-outs Entries`	Number of times the information advertised by neighbors has been removed from the neighbor table because the information TTL interval has expired.

Related Commands clear lldp statistics
show lldp statistics interface

show lldp statistics interface

This command displays the LLDP statistics (packet and event counters) for specified ports. If no port list is supplied, LLDP statistics for all ports are displayed.

Syntax

```
show lldp statistics interface [<port-list>]
```

Parameter	Description
<port-list>	The ports for which the statistics are to be shown.

Mode User Exec and Privileged Exec

Examples To display LLDP statistics information for all ports, use the command:

```
awplus# show lldp statistics interface
```

To display LLDP statistics information for ports 1.0.1 and 1.0.7, use the command:

```
awplus# show lldp statistics interface port1.0.1,port1.0.7
```

Output

Figure 66-8: Example output from the **show lldp statistics interface** command

```
awplus# show lldp statistics interface port1.0.1,port1.0.7

LLDP Packet and Event Counters:

port1.0.1
  Frames:    Out ................... 27
             In .................... 22
             In Errored ............ 0
             In Dropped ............ 0
  TLVs:      Unrecognized .......... 0
             Discarded ............. 0
  Neighbors: New Entries ........... 3
             Deleted Entries ....... 0
             Dropped Entries ....... 0
             Entry Age-outs ........ 0

port1.0.7
  Frames:    Out ................... 15
             In .................... 18
             In Errored ............ 0
             In Dropped ............ 0
  TLVs:      Unrecognized .......... 0
             Discarded ............. 0
  Neighbors: New Entries ........... 1
             Deleted Entries ....... 0
             Dropped Entries ....... 0
             Entry Age-outs ........ 0
```

Table 66-8: Parameters in the output of the **show lldp statistics interface** command

Parameter	Description
`Frames Out`	Number of LLDPDU frames transmitted.
`Frames In`	Number of LLDPDU frames received.
`Frames In Errored`	Number of invalid LLDPDU frames received.

Table 66-8: Parameters in the output of the **show lldp statistics interface**

Parameter	Description
`Frames In Dropped`	Number of LLDPDU frames received and discarded for any reason.
`TLVs Unrecognized`	Number of LLDP TLVs received that are not recognized but the TLV type is in the range of reserved TLV types.
`TLVs Discarded`	Number of LLDP TLVs discarded for any reason.
`Neighbors New Entries`	Number of times the information advertised by neighbors has been inserted into the neighbor table.
`Neighbors Deleted Entries`	Number of times the information advertised by neighbors has been removed from the neighbor table.
`Neighbors Dropped Entries`	Number of times the information advertised by neighbors could not be entered into the neighbor table because of insufficient resources.
`Neighbors Entry Age-outs Entries`	Number of times the information advertised by neighbors has been removed from the neighbor table because the information TTL interval has expired.

Related Commands

clear lldp statistics
show lldp statistics

show location

Use this command to display selected location information configured on the switch.

Syntax

```
show location {civic-location|coord-location|elin-location}

show location {civic-location|coord-location|elin-location}
    identifier {<civic-loc-id>|<coord-loc-id>|<elin-loc-id>}

show location {civic-location|coord-location|elin-location} interface
    <port-list>
```

Parameter	Description
`civic-location`	Display civic location information.
`coord-location`	Display coordinate location information.
`elin-location`	Display ELIN location information.
<civic-loc-id>	Civic address location identifier, in the range 1 to 4095.
<coord-loc-id>	Coordinate location identifier, in the range 1 to 4095.
<elin-loc-id>	ELIN location identifier, in the range 1 to 4095.
<port-list>	Ports to display information about.

Mode User Exec and Privileged Exec

Examples To display a civic address location configured on port1.0.1, use the command:

```
awplus# show location civic-location interface port1.0.1
```

Figure 66-9: Example output from the **show location** command

```
awplus# show location civic-location interface port1.0.1
Port      ID   Element Type           Element Value
----------------------------------------------------------------
1.0.1     1    Country                NZ
               City                   Christchurch
               Street-suffix          Avenue
               House-number           27
               Primary-road-name      Nazareth
```

To display coordinate location information configured on the identifier 1, use the command:

```
awplus# show location coord-location identifier 1
```

Figure 66-10: Example output from the **show location** command

```
awplus# show location coord-location identifier 1
  ID  Element Type              Element Value
--------------------------------------------------------------------
  1   Latitude Resolution       15 bits
      Latitude                  38.8986481130123138427734375 degrees
      Longitude Resolution      15 bits
      Longitude                 130.2323232293128967285156250 degrees
      Altitude Resolution       10 bits
      Altitude                  2.50000000 meters
      Map Datum                 WGS 84
```

The coordinate location information displayed may differ from the information entered because it is stored in binary format. For more information, see the location coord-location configuration command.

To display all ELIN location information configured on the switch, use the command:

```
awplus# show location elin-location
```

Figure 66-11: Example output from the show location command

```
awplus# show location elin-location
  ID  ELIN
-----------------------------------
  1   1234567890
  2   5432154321
```

Related Commands

location elin-location-id
location civic-location identifier
location civic-location configuration
location coord-location identifier
location coord-location configuration
location elin-location

Chapter 67: SMTP Commands

Command List

This chapter provides an alphabetical reference for commands used to configure SMTP.

For information about modifying or redirecting the output from **show** commands to a file, see "Controlling "show" Command Output" on page 1.41.

debug mail

This command turns on debugging for sending emails.

The **no** variant of this command turns off debugging for sending emails.

Syntax

```
debug mail
```

```
no debug mail
```

Mode Privileged Exec

Examples To turn on debugging for sending emails, use the command:

```
awplus# debug mail
```

To turn off debugging for sending emails, use the command:

```
awplus# no debug mail
```

Related Commands

delete mail
mail
mail from
mail smtpserver
show mail
show counter mail
undebug mail

delete mail

This command deletes mail from the queue.

Syntax

```
delete mail [mail-id <mail-id>|all]
```

Parameter	Description	
`mail-id`	Deletes a single mail from the mail queue.	
	`<mail-id>`	An unique mail ID number. Use the show mail command to display this for an item of mail.
`all`	Delete all the mail in the queue.	

Mode Privileged Exec

Examples To delete a unique mail item `20060912142356.1234` from the queue, use the command:

```
awplus# delete mail 20060912142356.1234
```

To delete all mail from the queue, use the command:

```
awplus# delete mail all
```

Related Commands

debug mail
mail
mail from
mail smtpserver
show mail

mail

This command sends an email using the SMTP protocol. If you specify a file the text inside the file is sent in the message body.

If you do not specify the **to**, **file**, or **subject** parameters, the CLI prompts you for the missing information.

Before you can send mail using this command, you must specify the sending email address using the mail from command and a mail server using the mail smtpserver command.

Syntax

```
mail [{to <to>|subject <subject>|file <filename>}]
```

Parameter	Description	
`to`	The email recipient.	
	<to>	Email address.
`subject`	Description of the subject of this email. Use quote marks when the subject text contains spaces.	
	<subject>	String.
`file`	File to insert as text into the message body.	
	<filename>	String.

Mode Privileged Exec

Example To send an email to `rei@nerv.com` with the subject `dummy plug configuration`, and with the message body inserted from the file `plug.conf` use the command:

```
awplus# mail rei@nerv.com subject dummy plug configuration
        filename plug.conf
```

Related Commands

debug mail
delete mail
mail from
mail smtpserver
show mail
show counter mail

mail from

This command sets an email address for the 'mail from' SMTP command. You must specify a sending email address with this command before you can send any email.

Syntax

```
mail from <from>
```

Parameter	Description
<from>	The email address that the mail is sent from.

Mode Global Configuration

Example To set the email address you are sending mail from to "kaji@nerv.com, use the command:

```
awplus(config)# mail from kaji@nerv.com
```

Related Commands

delete mail
mail
mail smtpserver
show mail

mail smtpserver

This command sets the IP address of the SMTP server that your device sends email to. You must specify a mail server with this command before you can send any email.

Syntax

```
mail smtpserver <ip-address>
```

Parameter	Description
<ip-address>	Internet Protocol (IP) Address for the mail server specified.

Mode Global Configuration

Example To specify a mail server at 192.168.0.1, use the command:

```
awplus# mail smtpserver 192.168.0.1
```

Related Commands

debug mail
delete mail
mail
mail from
show mail
show counter mail

show counter mail

This command displays the mail counters.

Syntax

```
show counter mail
```

Mode

User Exec and Privileged Exec

Output

Figure 67-1: Example output from the **show counter mail** command

```
Mail Client (SMTP) counters
Mails Sent                ......... 0
Mails Sent Fails          ......... 1
```

Table 67-1: Parameters in the output of the **show counter mail** command

Parameter	Description
`Mails Sent`	The number of emails sent successfully since the last device restart.
`Mails Sent Fails`	The number of emails the device failed to send since the last device restart.

Example

To show the emails in the queue use the command:

```
awplus# show counter mail
```

Related Commands

debug mail
delete mail
mail
mail from
show mail

show mail

This command displays the emails in the queue.

Syntax

```
show mail
```

Mode Privileged Exec

Example To display the emails in the queue use the command:

```
awplus# show mail
```

Related Commands

delete mail
mail
show counter mail

undebug mail

This command applies the functionality of the no debug mail command on page 67.2.

Chapter 68: RMON Introduction and Configuration

Introduction

The chapter describes the Remote Network MONitoring (RMON) service on the switch, and describes a configuration example showing how to set up an RMON alarm.

This RMON alarm configuration example described creates SNMP traps and log messages when the rate of receipt of Broadcast packets on a switch port exceeds a threshold, and creates SNMP traps and log messages when the rate of receipt of Broadcast packets on a switch drops below a lower threshold.

For detailed information about the commands used to configure RMON, see Chapter 69, RMON Commands

RMON is disabled by default in AlliedWare Plus^{TM}. No RMON alarms or events are configured.

Overview

The Remote Network MONitoring (RMON) MIB (RFC2819) was developed by the IETF to support monitoring and protocol analysis of LANs with a focus on Layer 1 and 2 information in networks. RMON is an industry standard that provides the functionality in network analyzers.

An RMON implementation operates in a client/server model. Monitoring devices (or 'probes') contain RMON agents that collect information and analyze packets. The probes are servers and the Network Management applications that communicate with them are clients. While agent configuration and data collection uses SNMP, RMON operates differently than SNMP systems:

- Probes have responsibility for data collection and processing, reducing SNMP traffic and reducing processing load for clients.
- Information is only transmitted to the management application when required, not polled.

RMON is mainly used for 'flow-based' monitoring, while SNMP is mainly used for 'device-based' management. RMON data collected deals mainly with traffic patterns on the network, and SNMP data collected usually deals with the status of individual devices on the network.

One disadvantage of flow based monitoring is that remote devices have much more of the management burden, and require more resources. AlliedWare Plus minimizes the management and resources burden by implementing a subset of the RMON MIB group to provide a minimal RMON agent implementation supporting statistics, history, alarms, and events.

The RMON groups supported in AlliedWare Plus^{TM} are:

- **Statistics** - collects ethernet statistics on a switch port, such as utilization and collisions.
- **History** - collects a history of ethernet statistics on a switch port.
- **Alarms** - monitor a MIB object for a specified interval, trigger an alarm at a specified value (the '**rising threshold**'), and resets the alarm at another value (the '**falling threshold**'). Alarms are used with events to trigger alarms, which generate logs or SNMP traps.
- **Events** - specify the action to take when an event is triggered by an alarm.
  The action of an event can generate a log or an SNMP trap.

RMON Configuration Example

This configuration example sets up an RMON alarm to create SNMP traps and log messages. This RMON alarm creates SNMP traps and log messages when the rate of receipt of Broadcast packets on a switch port exceeds a threshold, and creates SNMP traps and log messages when the rate of receipt of Broadcast packets on a switch port drops below a lower threshold.

Step 1: Set up an RMON collection on the switch port that is being monitored.

Use the following commands to configure this functionality:

```
          awplus# configure terminal

   awplus(config)# interface port1.0.4

awplus(config-if)# rmon collection stats 4
```

This will cause the software to build a table in which it stores statistics relating to the switch port.

Step 2: Define an RMON event that will be called by the Alarm when the thresholds are passed.

Create this as a 'trap and log' event, so that both an SNMP trap and a log message will be generated. The trap will be sent to the SNMP community named 'public'.

Use the following command to configure this functionality:

```
awplus(config-if)# rmon event 10 log trap public
```

Step 3: Create the RMON alarm.

Every 5 seconds, the alarm checks the broadcast packet counter in RMON collection stats 4. If the change in the value of that counter over the 5 second interval exceeds 5000 (1000 broadcasts per second), the alarm will trigger the event defined in step 2 above.

Additionally, when the rate broadcast falls below 500 broadcasts per 5 seconds, then the alarm will trigger the event defined in step 2 above again.

Use the below command to configure this functionality:

```
awplus(config-if)# rmon alarm 5 etherStatsBroadcastPkts.4
                   interval 5 delta rising-threshold 5000
                   event 10 falling-threshold 500 event 10
```

For the variable '`etherStatsBroadcastPkts.4`' in this command, note that '`.4`' refers to the index number of the RMON collection stats 4 as defined on `port1.0.4`.

So, '`etherStatsBroadcastPkts.4`' refers to 'Received broadcasts' in RMON collection stats 4. Further counters for RMON are defined in section 5 of RFC 1757.

Step 4: Enable RMON traps.

To ensure that the SNMP trap is sent, you need to enabled RMON traps, and you need to define a trap host in SNMP. Use the below commands to configure this functionality:

```
        awplus# configure terminal

awplus(config)# snmp-server

awplus(config)# snmp-server enable trap rmon

awplus(config)# snmp-server community public

awplus(config)# snmp-server host 192.168.2.254 version 2c
                public
```

Note that the resulting log message will be of the form listed below:

```
RMON [1024]: Alarm Index 5 alarm Rising Threshold 5000 alarm
Value 5117 alarm Rising event Index 10 event description
RMON_SNMP
```

Chapter 69: RMON Commands

Command List

This chapter provides an alphabetical reference for commands used to configure Remote Monitoring (RMON).

For an introduction to RMON and an RMON configuration example, see Chapter 68, RMON Introduction and Configuration

RMON is disabled by default in AlliedWare Plus^{TM}. No RMON alarms or events are configured.

For information about modifying or redirecting the output from **show** commands to a file, see "Controlling "show" Command Output" on page 1.41.

rmon alarm

Use this command to configure an RMON alarm to monitor the value of an SNMP object, and to trigger specified events when the monitored object crosses specified thresholds.

To specify the action taken when the alarm is triggered, use the event index of an event defined by the rmon event command.

Use the **no** variant of this command to remove the alarm configuration.

Note Only alarms for switch port interfaces, not for VLAN interfaces, can be configured.

Syntax

```
rmon alarm <alarm-index> <oid> interval <1-4294967295> {delta|
   absolute} rising-threshold <1-2147483647> event <rising-event-
   index> falling-threshold <1-2147483647> event <falling-event-
   index> [owner <owner>]
```

```
no rmon alarm <alarm-index>
```

Parameter	Description
`<alarm-index>`	<1-65535> Alarm entry index value.
`<oid>`	The variable SNMP MIB Object Identifier (OID) name to be monitored, in the format etherStatsEntry.field.<*stats-index*>. For example, etherStatsEntry.5.22 is the OID for the etherStatsPkts field in the etherStatsEntry table for the interface defined by the <*stats-index*> 22 in the rmon collection stats command.
`interval <1-4294967295>`	Polling interval in seconds.
`delta`	The RMON MIB alarmSampleType: the change in the monitored MIB object value between the beginning and end of the polling interval.
`absolute`	The RMON MIB alarmSampleType: the value of the monitored MIB object.
`rising-threshold <1-2147483647>`	Rising threshold value of the alarm entry in seconds.
`<rising-event-index>`	*<1-65535>* The event to be triggered when the monitored object value reaches the rising threshold value. This is an event index of an event specified by the rmon event command.
`falling-threshold <1-2147483647>`	Falling threshold value of the alarm entry in seconds.
`<falling-event-index>`	*<1-65535>* The event to be triggered when the monitored object value reaches the falling threshold value. This is an event index of an event specified by the rmon event command.
`owner <owner>`	Arbitrary owner name to identify the alarm entry.

Default By default, there are no alarms.

Mode Global Configuration

Usage Note that the SNMP MIB Object Identifier (OID) indicated in the command syntax with <*oid*> must be specified as a dotted decimal value with the form **etherStatsEntry.field.**<*stats-index*>.

Example To configure an alarm to monitor the change per minute in the etherStatsPkt value for interface 22 (defined by stats-index 22 in the rmon collection stats command), to trigger event 2 (defined by the rmon event command) when it reaches the rising threshold 400, and to trigger event 3 when it reaches the falling threshold 200, and identify this alarm as belonging to Maria, use the commands:

```
        awplus# configure terminal

awplus(config)# rmon alarm 229 etherStatsEntry.22.5 interval 60
                delta rising-threshold 400 event 2 falling-
                threshold 200 event 3 owner maria
```

Related Commands rmon collection stats
rmon event

rmon collection history

Use this command to create a history statistics control group to store a specified number of snapshots (buckets) of the standard RMON statistics for the switch port, and to collect these statistics at specified intervals. If there is sufficient memory available, then the device will allocate memory for storing the set of buckets that comprise this history control.

Use the **no** variant of this command to remove the specified history control configuration.

Note Only a history for switch port interfaces, not for VLAN interfaces, can be collected.

Syntax

```
rmon collection history <history-index> [buckets <1-65535>]
    [interval <1-3600>] [owner <owner>]

no rmon collection history <history-index>
```

Parameter	Description
`<history-index>`	<1-65535> A unique RMON history control entry index value.
`buckets <1-65535>`	Number of requested buckets to store snapshots. Default 50 buckets.
`interval <1-3600>`	Polling interval in seconds. Default 1800 second polling interval.
`owner <owner>`	Owner name to identify the entry.

Default The default interval is 1800 seconds and the default buckets is 50 buckets.

Mode Interface Configuration

Example

```
         awplus# configure terminal
 awplus(config)# interface port1.0.2
awplus(config-if)# rmon collection history 200 buckets 500
                   interval 600 owner herbert

         awplus# configure terminal
 awplus(config)# interface port1.0.2
awplus(config-if)# no rmon collection history 200
```

rmon collection stats

Use this command to enable the collection of RMON statistics on a switch port, and assign an index number by which to access these collected statistics.

Use the **no** variant of this command to stop collecting RMON statistics on this switch port.

Note Only statistics for switch port interfaces, not for VLAN interfaces, can be collected.

Syntax

```
rmon collection stats <collection-index> [owner <owner>]

no rmon collection stats <collection-index>
```

Parameter	Description
<collection-index>	*<1-65535>* Give this collection of statistics an index number to uniquely identify it. This is the index to use to access the statistics collected for this switch port.
owner *<owner>*	An arbitrary owner name to identify this statistics collection entry.

Default RMON statistics are not enabled by default.

Mode Interface Configuration

Example

```
          awplus# configure terminal
  awplus(config)# interface port1.0.3
awplus(config-if)# rmon collection stats 200 owner myrtle

          awplus# configure terminal
  awplus(config)# interface port1.0.3
awplus(config-if)# no rmon collection stats 200
```

rmon event

Use this command to create an event definition for a log or a trap or both. The event index for this event can then be referred to by the rmon alarm command.

Use the **no** variant of this command to remove the event definition.

Note Only the events for switch port interfaces, not for VLAN interfaces, can be collected.

Syntax

```
rmon event <event-index> [description <description>|owner <owner>|
    trap <trap>]

rmon event <event-index> [log [description <description>|
    owner <owner>|trap <trap>] ]

rmon event <event-index> [log trap [description <description>|
    owner <owner] ]

no rmon event <event-index>
```

Parameter	Description
<event-index>	*<1-65535>* Unique event entry index value.
log	Log event type.
trap	Trap event type.
log trap	Log and trap event type.
description *<description>*	Event entry description.
owner *<owner>*	Owner name to identify the entry.

Default No event is configured by default.

Mode Global Configuration

Example

```
        awplus# configure terminal
awplus(config)# rmon event 299 log description cond3 owner
                alfred
```

```
        awplus# configure terminal
awplus(config)# no rmon event 299
```

Related Commands rmon alarm

show rmon alarm

Use this command to display the alarms and threshold configured for the RMON probe.

Note Only the alarms for switch port interfaces, not for VLAN interfaces, can be shown.

Syntax `show rmon alarm`

Mode User Exec and Privileged Exec

Example

```
awplus# show rmon alarm
```

Related Commands rmon alarm

show rmon event

Use this command to display the events configured for the RMON probe.

Note Only the events for switch port interfaces, not for VLAN interfaces, can be shown.

Syntax `show rmon event`

Mode User Exec and Privileged Exec

Output Figure 69-1: Example output from the **show rmon event** command

```
awplus#sh rmon event
 event Index = 787
       Description TRAP
        Event type log & trap
         Event community name gopher
       Last Time Sent = 0
       Owner  RMON_SNMP

 event Index = 990
       Description TRAP
        Event type trap
         Event community name teabo
       Last Time Sent = 0
       Owner  RMON_SNMP
```

Note The following etherStats counters are not currently available for Layer 3 interfaces:

- etherStatsBroadcastPkts
- etherStatsCRCAlignErrors
- etherStatsUndersizePkts
- etherStatsOversizePkts
- etherStatsFragments
- etherStatsJabbers
- etherStatsCollisions
- etherStatsPkts64Octets
- etherStatsPkts65to127Octets
- etherStatsPkts128to255Octets
- etherStatsPkts256to511Octets
- etherStatsPkts512to1023Octets
- etherStatsPkts1024to1518Octets

Example

```
awplus# show rmon event
```

Related Commands rmon event

show rmon history

Use this command to display the parameters specified on all the currently defined RMON history collections on the device.

Note Only the history for switch port interfaces, not for VLAN interfaces, can be shown.

Syntax `show rmon history`

Mode User Exec and Privileged Exec

Output Figure 69-2: Example output from the **show rmon history** command

```
awplus#sh rmon history
 history index = 56
         data source ifindex = 4501
         buckets requested = 34
        buckets granted = 34
        Interval = 2000
        Owner Andrew

 history index = 458
         data source ifindex = 5004
         buckets requested = 400
        buckets granted = 400
        Interval = 1500
        Owner trev
========================================================
```

Note The following etherStats counters are not currently available for Layer 3 interfaces:

- etherStatsBroadcastPkts
- etherStatsCRCAlignErrors
- etherStatsUndersizePkts
- etherStatsOversizePkts
- etherStatsFragments
- etherStatsJabbers
- etherStatsCollisions
- etherStatsPkts64Octets
- etherStatsPkts65to127Octets
- etherStatsPkts128to255Octets
- etherStatsPkts256to511Octets
- etherStatsPkts512to1023Octets
- etherStatsPkts1024to1518Octets

Example

```
awplus# show rmon history
```

Related Commands rmon collection history

show rmon statistics

Use this command to display the current values of the statistics for all the RMON statistics collections currently defined on the device.

Note Only statistics for switch port interfaces, not for VLAN interfaces, can be shown.

Syntax `show rmon statistics`

Mode User Exec and Privileged Exec

Example

```
awplus# show rmon statistics
```

Output Figure 69-3: Example output from the **show rmon statistics** command

```
awplus#show rmon statistics
    rmon collection index 45
    stats->ifindex = 4501
    input packets 1279340, bytes 85858960, dropped 00, multicast packets 1272100
    output packets 7306090, bytes 268724, multicast packets 7305660 broadcast
packets 290
    rmon collection index 679
    stats->ifindex = 5013
    input packets 00, bytes 00, dropped 00, multicast packets 00
    output packets 8554550, bytes 26777324, multicast packets 8546690 broadcast
packets 7720
```

Note The following etherStats counters are not currently available for Layer 3 interfaces:

- etherStatsBroadcastPkts
- etherStatsCRCAlignErrors
- etherStatsUndersizePkts
- etherStatsOversizePkts
- etherStatsFragments
- etherStatsJabbers
- etherStatsCollisions
- etherStatsPkts64Octets
- etherStatsPkts65to127Octets
- etherStatsPkts128to255Octets
- etherStatsPkts256to511Octets
- etherStatsPkts512to1023Octets
- etherStatsPkts1024to1518Octets

Related Commands rmon collection stats

Chapter 70: Triggers Introduction

Introduction

This chapter provides information about the Trigger facility on this switch. For specific configuration examples, see Chapter 71, Triggers Configuration. For detailed descriptions of the commands used to configure triggers, see Chapter 72, Trigger Commands.

Trigger Facility

The Trigger facility provides a powerful mechanism for automatic and timed management of your device by automating the execution of commands in response to certain events. For example, you can use triggers to deactivate a service during the weekends, or to collect diagnostic information when the CPU usage is high.

A **trigger** is an ordered sequence of scripts that is executed when a certain event occurs. A **script** is a sequence of commands stored as a plaintext file on a file subsystem accessible to the device, such as Flash memory. Each trigger may reference multiple scripts and any script may be used by any trigger. When an event activates a trigger, the trigger executes the scripts associated with it in sequence. One script is executed completely before the next script begins. Various types of triggers are supported, each activated in a different way.

Configuring a Trigger

The following describes the general steps to configure a trigger. For specific configuration examples, see Chapter 71, Triggers Configuration.

Step 1: Create a configuration script

Create a configuration script with the commands you would like executed when the trigger conditions are met. To create the configuration script using the CLI, use the command:

```
awplus# edit [<filename>]
```

Alternatively, you can create a script on a PC then load it onto your device using the copy (URL) command.

Step 2: Enter the trigger configuration mode

You must be in the Global Configuration mode to reach the Trigger Configuration mode. Use the command:

```
awplus# configure terminal
```

To create a trigger, and enter its configuration mode, use the command:

```
awplus(config)# trigger <1-250>
```

Step 3: Set the trigger type

The trigger type determines how the trigger is activated. To set the trigger to activate:

- when a Secure Digital (SD) or Secure Digital High Capacity (SDHC) card is either inserted or removed, use the command:

```
awplus(config-trigger)# type card {in|out}
```

- when CPU usage reaches a certain level, use the command:

```
awplus(config-trigger)# type cpu <1-100> [up|down|any]
```

- when the link status of a particular interface changes, use the command:

```
awplus(config-trigger)# type interface <interface>
                        [up|down|any]
```

- when the RAM usage reaches a certain level, use the command:

```
awplus(config-trigger)# type memory <1-100> [up|down|any]
```

- periodically after a set number of minutes, use the command:

```
awplus(config-trigger)# type periodic <1-1440>
```

- when a ping poll identifies that a target device's status has changed, use the command:

```
awplus(config-trigger)# type ping-poll <1-100> {up|down}
```

- if your device reboots, use the command:

```
awplus(config-trigger)# type reboot
```

- when a stacking link goes up or down, use the command:

```
awplus(config-trigger)# type stack link {up|down}
```

- at a specific time of the day, use the command:

```
awplus(config-trigger)# type time <hh:mm>
```

Note that a combined limit of 10 triggers of the type periodic and type time can be configured. If you attempt to add more than 10 triggers the following error message is displayed:

```
% Cannot configure more than 10 triggers with the type time or
periodic
```

Step 4: Set the time and days that the trigger can activate on

By default triggers can activate at any time of the day, on all days. If you want your trigger to activate only during a specific time of the day, use the command:

```
awplus(config-trigger)# time {[after <hh:mm:ss>]
                        [before <hh:mm:ss>]}
```

If you want your trigger to activate only on a specific date, use the command:

```
awplus(config-trigger)# day <1-31> <month> <2000-2035>
```

If you want the trigger to activate only on specific days of the week, use the command:

```
awplus(config-trigger)# day <weekday>
```

Note that you can set either a specific date, or specific weekdays, but not both.

Step 5: Specify how often the trigger can activate

By default, triggers can activate an unlimited number of times, as long as the trigger conditions are met. To set a limit on the number of times a trigger can activate, use the command:

```
awplus(config-trigger)# repeat {forever|no|once|yes|
                        <1-4294967294>}
```

You device maintains two counters that track the number of times a trigger has activated. One counts the total number of times the trigger is activated and is only reset if the device restarts, or when the trigger is destroyed. The other counter tracks the permitted number of repetitions. To reset this counter, use the repeat command on page 72.7.

Step 6: Add the script to the trigger

You can add up to five scripts to the trigger. When a trigger is activated, it executes the scripts in sequence, with the lowest numbered script activated first. The first script runs to completion before the next script begins. To add a script, use the command:

```
awplus(config-trigger)# script <1-5> {<filename>}
```

Step 7: Specify a description for the trigger

Specify a description for the trigger, so that you can easily identify the trigger in show commands and log output. Use the command:

```
awplus(config-trigger)# description <description>
```

Step 8: Verify the trigger’s configuration

To check the configuration of the trigger, use the command:

```
awplus(config-trigger)# show trigger [<1-250>|counter|
                        full]
```

Troubleshooting Triggers

You can use the trigger diagnostic mode and trigger debugging to test your triggers and troubleshoot any issues.

Diagnostic mode is set per trigger. In this mode the trigger activates if its trigger conditions are met, but does not run any of its scripts. Your device generates a log message to indicate that the trigger was activated. To place a trigger in diagnostic mode, enter the trigger's configuration mode and use the command:

```
awplus(config-trigger)# test
```

To start debugging for triggers, use the command:

```
awplus(config-trigger)# debug trigger
```

This generates detailed messages about how your device is processing the trigger commands and activating the triggers.

Enabling and Disabling

Triggers are enabled by default. This allows the trigger to activate as soon as its trigger conditions are met. If you need to disable a trigger but do not want to delete the trigger, use the command:

```
awplus(config-trigger)# no active
```

To enable the trigger again, use the command:

```
awplus(config-trigger)# active
```

To delete the trigger, use the command:

```
awplus(config-trigger)# no trigger <1-250>
```

Chapter 71: Triggers Configuration

Introduction

The chapter describes how to configure triggers to:

- Restrict Internet Access
- Capture Unusual CPU and RAM Activity
- See Daily Statistics
- Turn Off Power to Port LEDs
- Capture Show Output and Save to a SD Card

For more information about triggers, see Chapter 70, Triggers Introduction. For detailed descriptions of the commands used to configure triggers, see Chapter 72, Trigger Commands.

Restrict Internet Access

In the following configuration the ACME company wants to restrict its employees from accessing popular video sharing websites as this is causing bandwidth problems during work hours. The ACME company is happy for workers to access the site after work hours.

Employee PCs at ACME are on vlan2. Two triggers with associated scripts are needed:

- Trigger 1 activates at 8.30am and runs a script called **shutdown.scp**. This script adds commands to restrict access to the specified sites
- Trigger 2 activates at 5.30pm and runs the script called **open.scp**. This script removes the configuration specified by shutdown.scp

1. Create the **shutdown.scp** script

   Create a configuration script using Access Control List commands to restrict users on vlan2 from accessing the specific sites.

2. Create the **open.scp** script

   Create a script to remove the ACL configuration specified in the **shutdown.scp** file.

3. Configure trigger 1

   To create trigger 1, use the commands:

```
             awplus# configure terminal
     awplus(config)# trigger 1
```

   Set the trigger to activate at 8:30am, by using the command:

```
awplus(config-trigger)# type time 08:30
```

   Set the trigger to activate on Monday, Tuesday, Wednesday, Thursday and Friday:

```
awplus(config-trigger)# day mon tue wed thur fri
```

   Add the script **shutdown.scp** to the trigger:

```
awplus(config-trigger)# script 1 shutdown.scp
```

Specify a helpful description, such as **Stops access to video sharing sites**. Use the command:

```
awplus(config-trigger)# description Stops access to video
                        sharing sites
```

Change to Global Configuration mode:

```
awplus(config-trigger)# exit
```

4. Configure trigger 2

   To create trigger 2, use the command:

```
awplus(config)# trigger 2
```

Set the trigger to activate at 5.30pm:

```
awplus(config-trigger)# type time 17:30
```

Set the trigger to activate on Monday, Tuesday, Wednesday, Thursday and Friday:

```
awplus(config-trigger)# day mon tue wed thur fri
```

Add the script **open.scp** to the trigger:

```
awplus(config-trigger)# script 1 open.scp
```

Specify a helpful description, such as **Access allowed to video sharing sites**. Use the command:

```
awplus(config-trigger)# description Access allowed to video
                        sharing sites
```

5. Verify the configuration

   To check the configuration of the triggers, use the commands:

```
awplus# show trigger 1
```

```
awplus# show trigger 2
```

Capture Unusual CPU and RAM Activity

The following configuration allows you to troubleshoot high CPU or RAM usage by the device. It uses two triggers to capture show output, and places this output in a file.

- Trigger 3 activates the script cpu-usage.scp when CPU usage is over 90% and can activate up to 5 times
- Trigger 4 activates the script ram-usage.scp when RAM usage is over 95%, and can activate up to 10 times

1. Create the cpu-usage.scp configuration script

   Create a script with the appropriate show command:

```
awplus# show cpu | redirect showcpu.txt
```

   The output of the **show cpu** command has been redirected into a file. It is not possible to display trigger script output on the terminal. Redirecting the command output to a file means it is available for later inspection.

   If the trigger activates on more than one occasion the contents of **showcpu.txt** will be overwritten with the latest output. To keep a full record for all activations of this trigger an ASH shell script can be added to the trigger to manage the output of the configuration script. For example:

```
#!/bin/ash
date >> showcpu_bkup.txt
cat showcpu.txt >> showcpu_bkup.txt
```

   This script concatenates that date and time of activation and the contents of **showcpu.txt** onto the end of the backup file **showcpu_bkup.txt** in flash memory.

   Note that the files may grow large accumulating data and consume available flash memory.

2. Create the ram-usage.scp configuration script

   Create a script with the appropriate show command:

```
awplus# show memory | redirect showmem.txt
```

   The output of the **show memory** command has been redirected into a file. It is not possible to display trigger script output on the terminal. Redirecting the command output to a file means it is available for later inspection.

   If the trigger activates on more than one occasion the contents of **showcpu.txt** will be overwritten with the latest output. To keep a full record for all activations of this trigger an ASH shell script can be added to the trigger to manage the output of the configuration script. For example:

```
#!/bin/ash
date >> showmem_bkup.txt
cat showmem.txt >> showmem_bkup.txt
```

   This script concatenates that date and time of activation and the contents of **showmem.scp** onto the end of the backup file **showmem_bkup.scp** in flash memory.

   Note that the files may grow large accumulating data and consume available flash memory.

3. Configure trigger 3

   To create trigger 3, use the commands:

```
               awplus# configure terminal
       awplus(config)# trigger 3
```

   Set the trigger to activate when CPU usage exceeds 80%:

```
awplus(config-trigger)# type cpu 90 up
```

   Add the script **cpu-usage.scp** to the trigger:

```
awplus(config-trigger)# script 1 cpu-usage.scp
```

   Return to Global Configuration mode:

```
awplus(config-trigger)# exit
```

4. Configure trigger 4

   To create trigger 4, use the command:

```
       awplus(config)# trigger 4
```

   Set the trigger to activate when RAM usage exceeds 95%:

```
awplus(config-trigger)# type cpu 95 up
```

   Add the script **cpu-usage.scp** to the trigger:

```
awplus(config-trigger)# script 1 ram-usage.scp
```

5. Verify the configuration

   To check the configuration of the triggers, use the command:

```
               awplus# show trigger 3
               awplus# show trigger 4
```

See Daily Statistics

The ACME company has recently set up QoS on its traffic to give traffic different priorities to the ISP. ACME wants to assess how much traffic is dropped with the QoS bandwidths set over the next week. To do this, they want to generate an hourly report on QoS traffic on the first day that this is implemented.

- Trigger 5 activates the script **qos-stats.scp** every 60 minutes.
  The trigger is set to only activate during work hours.

1. Create the **qos-stats.scp** script

   Create a configuration script with the appropriate show commands. You can either create the configuration script using the CLI with the edit command or create a script on a PC then load it onto your device using the copy (URL) command.

2. Configure trigger 5

   To create trigger 5, use the commands:

```
          awplus#  configure terminal
  awplus(config)#  trigger 5
```

   Set the trigger to activate periodically every 60 minutes:

```
awplus(config-trigger)#  type periodic 60
```

   Set the trigger to activate only during the hours of 8:00am and 6:00pm:

```
awplus(config-trigger)#  time after 8:00 before 18:00
```

   Add the script **qos-stats.scp** to the trigger:

```
awplus(config-trigger)#  script 1 qos-stats.scp
```

3. Verify the configuration

   To check the configuration of the trigger, use the command:

```
          awplus#  show trigger 5
```

Turn Off Power to Port LEDs

The following configuration allows you to conserve power by using the eco-friendly feature to turn off power to the port LEDs during non-work hours.

- Trigger 6 activates at 5.30pm and runs a script called **LEDoff.scp**. This script adds commands to turn off power to all the port LEDs
- Trigger 7 activates at 8.30am and runs the script called **LEDon.scp**. This script removes the configuration specified by **LEDoff.scp**

1. Create the **LEDoff.scp** script

   Create a configuration script with the commands that are executed when the trigger conditions are met. You can either create the configuration script using the CLI with the edit command or create a script on a PC then load it onto your device using the copy (URL) command. The configuration script for this example is:

```
!
enable
configure terminal
ecofriendly led
exit
exit
!
```

2. Create the **LEDon.scp** script

   Create a script to remove the configuration specified in the **LEDoff.scp** file. The configuration script for this example is:

```
!
enable
configure terminal
no ecofriendly led
exit
exit
!
```

3. Configure trigger 6

   To create trigger 6, use the commands:

```
        awplus# configure terminal
awplus(config)# trigger 6
```

   Set the trigger to activate at 5:30pm, by using the command:

```
awplus(config-trigger)# type time 17:30
```

   Set the trigger to activate on Monday, Tuesday, Wednesday, Thursday and Friday:

```
awplus(config-trigger)# day mon tue wed thur fri
```

   Add the script **LEDoff.scp** to the trigger:

```
awplus(config-trigger)# script 1 powershutdown.scp
```

Specify a helpful description, such as **Shutdown power to LEDs**. Use the command:

```
awplus(config-trigger)# description Shutdown power to LEDs
```

Change to Global Configuration mode:

```
awplus(config-trigger)# exit
```

4. Configure trigger 7

   To create trigger 7, use the command:

```
        awplus(config)# trigger 9
```

Set the trigger to activate at 8.30am:

```
awplus(config-trigger)# type time 08:30
```

Set the trigger to activate on Monday, Tuesday, Wednesday, Thursday and Friday:

```
awplus(config-trigger)# day mon tue wed thur fri
```

Add the script **LEDon.scp** to the trigger:

```
awplus(config-trigger)# script 1 poweropen.scp
```

Specify a helpful description, such as **Turn on power to LEDs**. Use the command:

```
awplus(config-trigger)# description Turn on power to LEDs
```

5. Verify the configuration

   To check the configuration of the triggers, use the commands:

```
                awplus# show trigger 6
                awplus# show trigger 7
```

Capture Show Output and Save to a SD Card

The following configuration allows you to automatically capture output from the show tech-support command when a USB storage device is inserted into the switch. It uses a script called by the USB storage device to trigger to capture the **show tech-support** output and places this output in a file on the USB storage device.

- Trigger 8 activates the script **shtech-sup.scp** when a USB storage device is inserted in the switch

1. Create the **shtech-sup.scp** script

   Create a configuration script with the commands that are executed when the trigger conditions are met. You can either create the configuration script using the CLI with the edit command or create a script on a PC then load it onto your device using the copy (URL) command. The configuration script for this example is:

```
!
enable
show tech-support outfile card:support.txt.gz
exit
end
!
```

2. Configure trigger 8

   To create trigger 8, use the commands:

```
              awplus# configure terminal
      awplus(config)# trigger 8
```

   Set the trigger to activate on the insertion of a USB storage device:

```
awplus(config-trigger)# type usb in
```

   Add the script **shtech-sup.scp** to the trigger:

```
awplus(config-trigger)# script 1 shtech-sup.scp
```

3. Verify the configuration

   To check the configuration of the triggers, use the command:

```
              awplus# show trigger 8
```

Capture Show Output and Save to a USB Storage Device

The following configuration allows you to automatically capture output from the show tech-support command when a USB storage device is inserted into the switch. It uses a script called by the USB storage device trigger to capture the **show tech-support** output and places this output in a file on the USB storage device.

- Trigger 9 activates the script **shtech-sup.scp** when an USB storage device is inserted in the switch

1. Create the **shtech-sup.scp** script

   Create a configuration script with the commands that are executed when the trigger conditions are met. You can either create the configuration script using the CLI with the edit command or create a script on a PC then load it onto your device using the copy (URL) command. The configuration script for this example is:

```
!
enable
show tech-support outfile usb:support.txt.gz
exit
end
!
```

2. Configure trigger 9

   To create trigger 9, use the commands:

```
awplus# configure terminal
awplus(config)# trigger 9
```

   Set the trigger to activate on the insertion of a USB storage device:

```
awplus(config-trigger)# type usb in
```

   Add the script **shtech-sup.scp** to the trigger:

```
awplus(config-trigger)# script 1 shtech-sup.scp
```

3. Verify the configuration

   To check the configuration of the triggers, use the command:

```
awplus# show trigger 9
```

Load a Release File From a USB Storage Device

The following configuration allows you to automatically load a release file from a USB storage device into Flash memory when a USB storage device is inserted into the switch. It uses a script called by the USB trigger to load the release file from the USB storage device.

Note that you can only specify that the release file is on a USB storage device if there is a backup release file already specified in Flash. See the boot system command for further information.

Caution **Anyone with physical access to the switch and who knows the name of the release file loaded by the trigger could insert a USB storage device and overwrite the boot configuration in Flash memory.**

■ Trigger 11 activates the script **copy.scp** when a USB storage device is inserted in the switch

1. Create the **copy.scp** script

   Create a configuration script with the commands that are executed when the trigger conditions are met. You can either create the configuration script using the CLI with the edit command or create a script on a PC then load it onto your device using the copy (URL) command. The configuration script for this example is:

```
!
enable
copy usb flash x510-5.4.2A.rel
wait 5
configure terminal
boot system x510-5.4.2A.rel
exit
end
!
```

2. Configure trigger 11

   To create trigger 11, use the commands:

```
        awplus# configure terminal
awplus(config)# trigger 11
```

   Set the trigger to activate on the insertion of a USB storage device:

```
awplus(config-trigger)# type usb in
```

   Add the script **copy.scp** to the trigger:

```
awplus(config-trigger)# script 1 copy.scp
```

   Specify a helpful description, such as **Load a release file**. Use the command:

```
awplus(config-trigger)# description Load a release file
```

After a USB storage device has been inserted in the switch, use the following two steps to check the trigger and current boot configuration details.

1. Verify the trigger configuration

   To check the configuration of the trigger, use the command:

```
awplus# show trigger 11
```

   Example output from this command is shown below:

```
awplus#show trigger 11
Trigger Configuration Details
------------------------------------------------------------
Trigger ..................... 11
Description ................. Load a release file
Type and details ............ USB (in)
Days ........................ smtwtfs
After ....................... 00:00:00
Before ...................... 23:59:59
Active ...................... Yes
Test ........................ No
Trap ........................ Yes
Repeat ...................... Continuous
Modified .................... Wed Sep 15 16:25:33 2010
Number of activations ....... 1
Last activation ............. Wed Sep 15 16:26:49 2010
Number of scripts ........... 1
  1. copy.scp
  2. <not configured>
  3. <not configured>
  4. <not configured>
  5. <not configured>
------------------------------------------------------------
```

2. Display the current boot configuration

   To display the current boot configuration, use the command:

```
awplus# show boot
```

   Example output from this command is shown below:

```
awplus#show boot
Boot configuration
------------------------------------------------------------
Current software   : x510-5.4.2A.rel
Current boot image : flash:/x510-5.4.2A.rel
Backup  boot image : flash:/x510-5.4.2A.rel
Default boot config: flash:/default.cfg
Current boot config: flash:/atplab.cfg (file exists)
Backup  boot config: flash:/default.cfg (file exists)
```

Chapter 72: Trigger Commands

Command List

This chapter provides an alphabetical reference for commands used to configure Triggers. For more information, see Chapter 70, Triggers Introduction and Chapter 71, Triggers Configuration.

For information about modifying or redirecting the output from **show** commands to a file, see "Controlling "show" Command Output" on page 1.41.

active (trigger)

This command enables a trigger. This allows the trigger to activate when its trigger conditions are met.

The **no** variant of this command disables a trigger. While in this state the trigger cannot activate when its trigger conditions are met.

Syntax

```
active
```

```
no active
```

Mode Trigger Configuration

Examples To enable trigger 172, so that it can activate when its trigger conditions are met, use the commands:

```
               awplus# configure terminal
       awplus(config)# trigger 172
awplus(config-trigger)# active
```

To disable trigger 182, preventing it from activating when its trigger conditions are met, use the commands:

```
               awplus# configure terminal
       awplus(config)# trigger 182
awplus(config-trigger)# no active
```

Related Commands show trigger
trigger

day

This command specifies the days or date that the can trigger activate on. You can specify either:

- A specific date
- A specific day of the week
- A list of days of the week
- every day

By default, the trigger can activate on any day.

Syntax

```
day every-day

day <1-31> <month> <2000-2035>

day <weekday>
```

Parameter	Description
`every-day`	Sets the trigger so that it can activate on any day.
<1-31>	Day of the month the trigger is permitted to activate on.
<month>	Sets the month that the trigger is permitted to activate on. Valid keywords are: **january, february, march, april, may, june, july, august, september, october, november**, and **december**.
<2000-2035>	Sets the year that the trigger is permitted to activate in.
<weekday>	Sets the days of the week that the trigger can activate on. You can specify one or more week days in a space separated list. Valid keywords are: **monday, tuesday, wednesday, thursday, friday, saturday**, and **sunday**.

Mode Trigger Configuration

Usage For example trigger configurations that use the **day** command, see "Restrict Internet Access" on page 71.2 and "Turn Off Power to Port LEDs" on page 71.7.

Examples To permit trigger 55 to activate on the `1 Jun 2010`, use the commands:

```
             awplus# configure terminal
     awplus(config)# trigger 55
awplus(config-trigger)# day 1 Jun 2010
```

To permit trigger 12 to activate on a Mondays, Wednesdays and Fridays, use the commands:

```
             awplus# configure terminal
     awplus(config)# trigger 12
awplus(config-trigger)# day monday wednesday friday
```

Related Commands show trigger
trigger

debug trigger

This command enables trigger debugging. This generates detailed messages about how your device is processing the trigger commands and activating the triggers.

The **no** variant of this command disables trigger debugging.

Syntax

```
debug trigger
```

```
no debug trigger
```

Mode Privilege Exec

Examples To start trigger debugging, use the command:

```
awplus# debug trigger
```

To stop trigger debugging, use the command:

```
awplus# no trigger
```

Related Commands

show trigger
test
trigger
undebug trigger

description (trigger)

This command adds an optional description to help you identify the trigger. This description is displayed in show command outputs and log messages.

The **no** variant of this command removes a trigger's description. The show command outputs and log messages stop displaying a description for this trigger.

Syntax

```
description <description>

no description
```

Parameter	Description
<description>	A word or phrase that uniquely identifies this trigger or its purpose. Valid characters are any printable character and spaces, up to a maximum of 40 characters.

Mode Trigger Configuration

Examples To give trigger 240 the description `daily status report`, use the commands:

```
                awplus# configure terminal
        awplus(config)# trigger 240
awplus(config-trigger)# description daily status report
```

To remove the description from trigger 36, use the commands:

```
                awplus# configure terminal
        awplus(config)# trigger 36
awplus(config-trigger)# no description
```

Related Commands

show trigger
test
trigger

repeat

This command specifies the number of times that a trigger is permitted to activate. This allows you to specify whether you want the trigger to activate:

- only the first time that the trigger conditions are met
- a limited number of times that the trigger conditions are met
- an unlimited number of times

Once the trigger has reached the limit set with this command, the trigger remains in your configuration but cannot be activated. Use the **repeat** command again to reset the trigger so that it is activated when its trigger conditions are met.

By default, triggers can activate an unlimited number of times. To reset a trigger to this default, specify either **yes** or **forever**.

Syntax

```
repeat {forever|no|once|yes|<1-4294967294>}
```

Parameter	Description
`yes|forever`	The trigger repeats indefinitely, or until disabled.
`no|once`	The trigger activates only once.
<1-4292967294>	The trigger repeats the set number of times.

Mode Trigger Configuration

Examples To allow trigger 21 to activate only once, use the commands:

```
               awplus# configure terminal
        awplus(config)# trigger 21
awplus(config-trigger)# repeat no
```

To allow trigger 22 to activate an unlimited number of times whenever its trigger conditions are met, use the commands:

```
               awplus# configure terminal
        awplus(config)# trigger 22
awplus(config-trigger)# repeat forever
```

To allow trigger 23 to activate only the first 10 times the conditions are met, use the commands:

```
               awplus# configure terminal
        awplus(config)# trigger 23
awplus(config-trigger)# repeat 10
```

Related Commands show trigger
trigger

script

This command specifies one or more scripts that are to be run when the trigger activates. You can add up to five scripts to a single trigger.

The sequence in which the trigger runs the scripts is specified by the number you set before the name of the script file. One script is executed completely before the next script begins.

Scripts may be either ASH shell scripts, indicated by a **.sh** filename extension suffix, or AlliedWare Plus™ scripts, indicated by a **.scp** filename extension suffix. AlliedWare Plus™ scripts only need to be readable.

The **no** variant of this command removes one or more scripts from the trigger's script list. The scripts are identified by either their name, or by specifying their position in the script list. The **all** parameter removes all scripts from the trigger.

Syntax

```
script <1-5> {<filename>}
```

```
no script {<1-5>|<filename>|all}
```

Parameter	Description
<1-5>	The position of the script in execution sequence. The trigger runs the lowest numbered script first.
<filename>	The path to the script file.

Mode Trigger Configuration

Examples To configure trigger 71 to run the script flash:/cpu_trig.sh in position 3 when the trigger activates, use the commands:

```
                awplus# configure terminal
        awplus(config)# trigger 71
awplus(config-trigger)# script 3 flash:/cpu_trig.sh
```

To configure trigger 99 to run the scripts **flash:reconfig.scp**, **flash:cpu_trig.sh** and **flash:email.scp** in positions 2, 3 and 5 when the trigger activates, use the following commands:

```
                awplus# configure terminal
        awplus(config)# trigger 99
awplus(config-trigger)# script 2 flash:/reconfig.scp 3 flash:/
                        cpu_trig.sh 5 flash:/email.scp
```

To remove the scripts 1, 3 and 4 from trigger 71's script list, use the commands:

```
                awplus# configure terminal
        awplus(config)# trigger 71
awplus(config-trigger)# no script 1 3 4
```

To remove the script flash:/cpu_trig.sh from trigger 71's script list, use the commands:

```
             awplus# configure terminal
     awplus(config)# trigger 71
awplus(config-trigger)# no script flash:/cpu_trig.sh
```

To remove all the scripts from trigger 71's script list, use the commands:

```
             awplus# configure terminal
     awplus(config)# trigger 71
awplus(config-trigger)# no script all
```

Related Commands show trigger
trigger

show running-config trigger

This command displays the current running configuration of the trigger utility.

Syntax

```
show running-config trigger
```

Mode

Privileged Exec

Example

To display the current configuration of the trigger utility, use the command:

```
awplus# show running-config trigger
```

Output

Figure 72-1: Example output from the **show running-config trigger** command

```
trigger 1
 type card in
trigger 2
 type card out
!
```

Related Commands

show trigger

show trigger

This command displays configuration and diagnostic information about the triggers configured on the device. Specify the **show trigger** command without any options to display a summary of the configuration of all triggers.

Syntax `show trigger [<1-250>|counter|full]`

Parameter	Description
`<1-250>`	Displays detailed information about a specific trigger, identified by its trigger ID.
`counter`	Displays statistical information about all triggers.
`full`	Displays detailed information about all triggers.

Mode Privileged Exec

Example To get summary information about all triggers, use the following command:

```
awplus# show trigger
```

Figure 72-2: Example output from the **show trigger** command

```
awplus#show trigger
TR# Type & Details       Name                    Ac Te Tr Repeat     #Scr Days/Date
---------------------------------------------------------------------------------
001 Card (in)                                    Y  N  Y  Continuous  0   smtwtfs
002 Card (out)                                   Y  N  Y  Continuous  0   smtwtfs
003 CPU (80% any)        Busy CPU                Y  N  Y  5           1   smtwtfs
005 Periodic (30 min)    Regular status check    Y  N  N  Continuous  1   -mtwtf-
007 Memory (85% up)      High mem usage          Y  N  Y  8           1   smtwtfs
011 Time (00:01)         Weekend access          Y  N  Y  Continuous  1   ------s
013 Reboot                                       Y  N  Y  Continuous  2   smtwtfs
017 Interface (vlan1 ... Change config for...    Y  N  Y  Once        1   2-apr-2008
019 Ping-poll (5 up)     Connection to svr1      Y  N  Y  Continuous  1   smtwtfs
---------------------------------------------------------------------------------
```

Table 72-1: Parameters in the output of the **show trigger** command

Parameter	Description
`TR#`	Trigger identifier (ID).
`Type & Details`	The trigger type, followed by the trigger details in brackets.
`Name`	Descriptive name of the trigger configured with the description (trigger) command.
`Ac`	Whether the trigger is active (Y), or inactive (N).
`Te`	Whether the trigger is in test mode (Y) or not (N).
`Tr`	Whether or not the trigger is enabled to send SNMP traps. See the trap command.

Table 72-1: Parameters in the output of the **show trigger** command(cont.)

Parameter	Description
`Repeat`	Whether the trigger repeats continuously, and if not, the configured repeat count for the trigger. To see the number of times a trigger has activated, use the **show trigger <1-250>** command.
`#Scr`	Number of scripts associated with the trigger.
`Days/Date`	Days or date when the trigger may be activated. For the days options, the days are shown as a seven character string representing Sunday to Saturday. A hyphen indicates days when the trigger cannot be activated.

To display detailed information about trigger 3, use the command:

```
awplus# show trigger 3
```

Figure 72-3: Example output from the **show trigger** command for a specific trigger

```
awplus#show trigger 3
Trigger Configuration Details
------------------------------------------------------------
Trigger ..................... 1
Description ................. display cpu usage when pass 80%
Type and details ............ CPU (80% up)
Days ........................ 26-nov-2007
After ....................... 00:00:00
Before ...................... 23:59:59
Active ...................... Yes
Test ........................ No
Trap ........................ Yes
Repeat ...................... 123 (0)
Modified .................... Tue Dec 20 02:26:03 1977
Number of activations ....... 0
Last activation ............. not activated
Number of scripts ........... 1
  1. shocpu.scp
  2. <not configured>
  3. <not configured>
  4. <not configured>
  5. <not configured>
------------------------------------------------------------
```

To display detailed information about all triggers, use the command:

```
awplus# show trigger full
```

Figure 72-4: Example output from the **show trigger full** command

```
awplus#show trigger full
Trigger Configuration Details
------------------------------------------------------------
 Trigger ..................... 1
 Description ................. <no description>
 Type and details ............ Card (in)
 Days ........................ smtwtfs
 After ....................... 00:00:00
 Before ...................... 23:59:59
 Active ...................... Yes
 Test ........................ No
 Trap ........................ Yes
 Repeat ...................... Continuous
 Modified .................... Fri Sep  3 14:45:56 2010
 Number of activations ....... 0
 Last activation ............. not activated
 Number of scripts ........... 0
   1. <not configured>
   2. <not configured>
   3. <not configured>
   4. <not configured>
   5. <not configured>

 Trigger ..................... 2
 Description ................. <no description>
 Type and details ............ Card (out)
 Days ........................ smtwtfs
 After ....................... 00:00:00
 Before ...................... 23:59:59
 Active ...................... Yes
 Test ........................ No
 Trap ........................ Yes
 Repeat ...................... Continuous
 Modified .................... Fri Sep  3 14:45:56 2010
 Number of activations ....... 0
 Last activation ............. not activated
 Number of scripts ........... 0
   1. <not configured>
   2. <not configured>
   3. <not configured>
   4. <not configured>
   5. <not configured>

Trigger ..................... 3
Description ................. Busy CPU
Type and details ............ CPU (80% up)
Days ........................ smtwtfs
Active ...................... Yes
Test ........................ No
Trap ........................ Yes
Repeat ...................... Continuous
Modified .................... Fri Feb 2 17:05:16 2007
Number of activations ....... 0
Last activation ............. not activated
Number of scripts ........... 2
  1. flash:/cpu_alert.sh
  2. flash:/reconfig.scp
  3. <not configured>
  4. <not configured>
  5. <not configured>
------------------------------------------------------------
```

Table 72-2: Parameters in the output of the **show trigger full** and **show trigger** commands for a specific trigger

Parameter	Description
`Trigger`	The ID of the trigger.
`Description`	Descriptive name of the trigger.
`Type and details`	The trigger type and its activation conditions.
`Days`	The days on which the trigger is permitted to activate.
`Date`	The date on which the trigger is permitted to activate. Only displayed if configured, in which case it replaces "Days".
`Active`	Whether or not the trigger is permitted to activate.
`Test`	Whether or not the trigger is operating in diagnostic mode.
`Trap`	Whether or not the trigger is enabled to send SNMP traps.
`Repeat`	Whether the trigger repeats an unlimited number of times (Continuous) or for a set number of times. When the trigger can repeat only a set number of times, then the number of times the trigger has been activated is displayed in brackets.
`Modified`	The date and time of the last time that the trigger was modified.
`Number of activations`	Number of times the trigger has been activated since the last restart of the device.
`Last activation`	The date and time of the last time that the trigger was activated.
`Number of scripts`	How many scripts are associated with the trigger, followed by the names of the script files in the order in which they run.

To display counter information about all triggers use the command:

```
awplus# show trigger counter
```

Figure 72-5: Example output from the **show trigger counter** command

```
awplus#show trigger counter
Trigger Module Counters
------------------------------------------------------
Trigger activations ........................... 0
Time triggers activated today ................. 0
Periodic triggers activated today ............. 0
Interface triggers activated today ............ 0
Resource triggers activated today ............. 0
Reboot triggers activated today ............... 0
Ping-poll triggers activated today ............ 0
Stack master fail triggers activated today .... 0
Stack member triggers activated today ......... 0
Stack xem-stk triggers activated today ........ 0
------------------------------------------------------
```

Table 72-3: Parameters in the output of the **show trigger counter** command

Parameter	Description
`Trigger activations`	Number of times a trigger has been activated.
`Time triggers activated today`	Number of times a time trigger has been activated today.
`Periodic triggers activated today`	Number of times a periodic trigger has been activated today.
`Interface triggers activated today`	Number of times an interface trigger has been activated today.
`Resource triggers activated today`	Number of times a CPU or memory resource trigger has been activated today.
`Ping-poll triggers activated today`	Number of times a ping-poll trigger has been activated today.

Related Commands trigger

test

This command puts the trigger into a diagnostic mode. In this mode the trigger may activate but when it does it will not run any of the trigger's scripts. A log message will be generated to indicate when the trigger has been activated.

The **no** variant of this command takes the trigger out of diagnostic mode, restoring normal operation. When the trigger activates the scripts associated with the trigger will be run, as normal.

Syntax

```
test
```

```
no test
```

Mode Trigger Configuration

Examples To put trigger 5 into diagnostic mode, where no scripts will be run when the trigger activates, use the commands:

```
               awplus#  configure terminal
       awplus(config)#  trigger 5
awplus(config-trigger)#  test
```

To take trigger 205 out of diagnostic mode, restoring normal operation, use the commands:

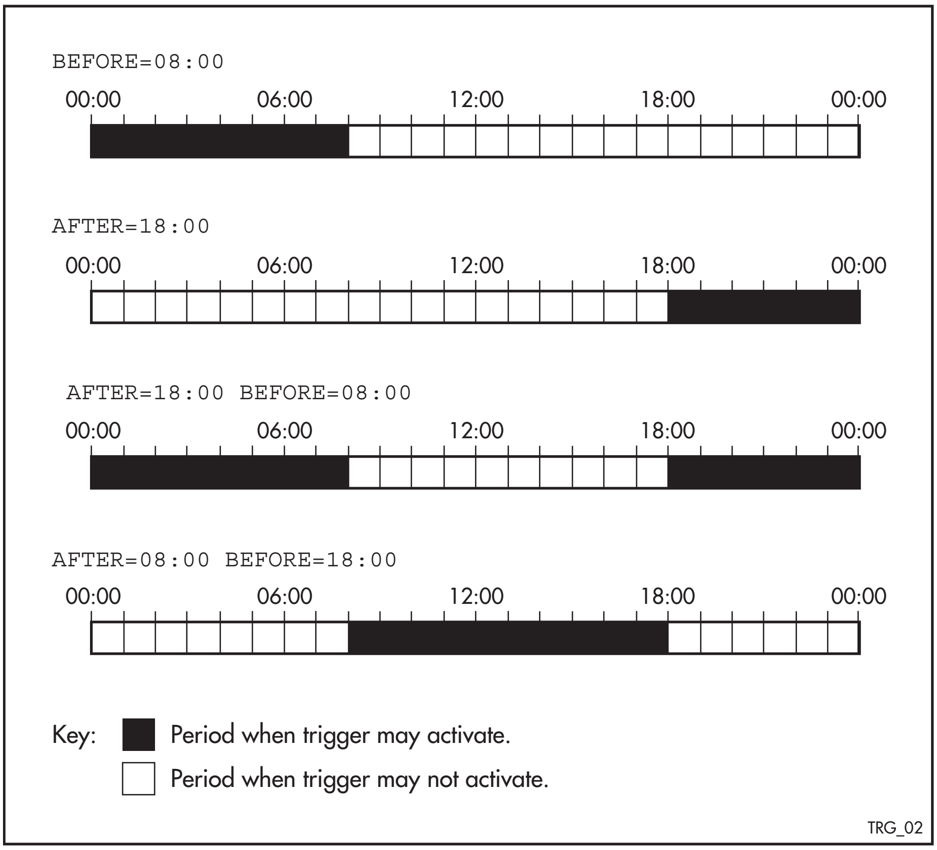
```
               awplus#  configure terminal
       awplus(config)#  trigger 205
awplus(config-trigger)#  no test
```

Related Commands show trigger
trigger

time (trigger)

This command specifies the time of day when the trigger is permitted to activate. The after parameter specifies the start of a time period that extends to midnight during which trigger may activate. By default the value of this parameter is 00:00:00 (am); that is, the trigger may activate at any time. The **before** parameter specifies the end of a time period beginning at midnight during which the trigger may activate. By default the value of this parameter is 23:59:59; that is, the trigger may activate at any time. If the value specified for **before** is later than the value specified for **after**, a time period from "after" to "before" is defined, during which the trigger may activate. This command is not applicable to time triggers (**type time**).

The following figure illustrates how the **before** and **after** parameters operate.



Syntax `time {[after <hh:mm:ss>] [before <hh:mm:ss>]}`

Parameter	Description
`after` *<hh:mm:ss>*	The earliest time of day when the trigger may be activated.
`before` *<hh:mm:ss>*	The latest time of day when the trigger may be activated.

Mode Trigger Configuration

Usage For example trigger configurations that use the **time (trigger)** command, see “Restrict Internet Access” on page 71.2 and “Turn Off Power to Port LEDs” on page 71.7.

Examples To allow trigger 63 to activate between midnight and 10:30am, use the commands:

```
              awplus#  configure terminal
      awplus(config)#  trigger 63
awplus(config-trigger)#  time before 10:30:00
```

To allow trigger 64 to activate between 3:45pm and midnight, use the commands:

```
              awplus#  configure terminal
      awplus(config)#  trigger 64
awplus(config-trigger)#  time after 15:45:00
```

To allow trigger 65 to activate between 10:30am and 8:15pm, use the commands:

```
              awplus#  configure terminal
      awplus(config)#  trigger 65
awplus(config-trigger)#  time after 10:30:00 before 20:15:00
```

Related Commands show trigger
trigger

trap

This command enables the specified trigger to send SNMP traps.

Use the **no** variant of this command to disable the sending of SNMP traps from the specified trigger.

Syntax

```
trap
```

```
no trap
```

Default SNMP traps are enabled by default for all defined triggers.

Mode Trigger Configuration

Usage You must configure SNMP before using traps with triggers. See the following SNMP chapters:
Chapter 62, SNMP Introduction
Chapter 63, SNMP Commands
Chapter 64, SNMP MIBs

Since SNMP traps are enabled by default for all defined triggers, a common usage will be for the **no** variant of this command to disable SNMP traps from a specified trap if the trap is only periodic. Refer in particular to AT-TRIGGER-MIB for further information about the relevant SNMP MIB.

Examples To enable SNMP traps to be sent from trigger 5, use the commands:

```
               awplus# configure terminal
       awplus(config)# trigger 5
awplus(config-trigger)# trap
```

To disable SNMP traps being sent from trigger 205, use the commands:

```
               awplus# configure terminal
       awplus(config)# trigger 205
awplus(config-trigger)# no trap
```

Related Commands trigger
show trigger

trigger

This command is used to access the Trigger Configuration mode for the specified trigger. Once Trigger Configuration mode has been entered the trigger type information can be configured and the trigger scripts and other operational parameters can be specified. At a minimum the trigger type information must be specified before the trigger can become active.

The **no** variant of this command removes a specified trigger and all configuration associated with it.

Syntax

```
trigger <1-250>

no trigger <1-250>
```

Parameter	Description
<1-250>	A trigger ID.

Mode Global Configuration

Examples To enter trigger configuration mode for trigger 12 use the command:

```
awplus# trigger 12
```

To completely remove all configuration associated with trigger 12, use the command:

```
awplus# no trigger 12
```

Related Commands show trigger
trigger activate

trigger activate

This command is used to manually activate a specified trigger from the Privileged Exec mode, which has been configured with the **trigger** command from the Global Configuration mode.

Syntax

```
trigger activate <1-250>
```

Parameter	Description
<1-250>	A trigger ID.

Mode Privileged Exec

Usage This command manually activates a trigger without the normal trigger conditions being met.

The trigger is activated even if it is configured as inactive. The scripts associated with the trigger will be executed even if the trigger is in the diagnostic test mode.

Triggers activated manually do not have their repeat counts decremented or their 'last triggered' time updated, and do not result in updates to the '[type] triggers today' counters.

Example To manually activate trigger 12 use the command:

```
awplus# trigger activate 12
```

Related Commands show trigger
trigger

type card

Use this command to configure a trigger that activates on either the removal or the insertion of a Secure Digital (SD) or Secure Digital High Capacity (SDHC) card.

Syntax

```
type card {in|out}
```

Parameter	Description
`in`	Trigger activates on insertion of a card.
`out`	Trigger activates on removal of a card.

Mode Trigger Configuration

Usage Card triggers cannot execute script files from a card.

In a VCStack configuration, card triggers are activated on the master for either the insertion or removal of a card on the master only.

For example trigger configurations that use the **type card** command, see "Capture Show Output and Save to a SD Card" on page 71.9.

Examples To configure `trigger 1` to activate on the insertion of a card, use the commands:

```
          awplus# configure terminal
  awplus(config)# trigger 1
awplus(config-trigger)# type card in
```

Related Commands trigger
show running-config trigger
show trigger

type cpu

This command configures a trigger to activate based on CPU usage level. Selecting the **up** option causes the trigger to activate when the CPU usage exceeds the specified usage level. Selecting the **down** option causes the trigger to activate when CPU usage drops below the specified usage level. Selecting **any** causes the trigger to activate in both situations. The default is **any**.

Syntax

```
type cpu <1-100> [up|down|any]
```

Parameter	Description
`<1-100>`	The percentage of CPU usage at which to trigger.
`up`	Activate when CPU usage exceeds the specified level.
`down`	Activate when CPU usage drops below the specified level
`any`	Activate when CPU usage passes the specified level in either direction

Mode Trigger Configuration

Usage For an example trigger configuration that uses the **type cpu** command, see "Capture Unusual CPU and RAM Activity" on page 71.4.

Examples To configure trigger 28 to be a CPU trigger that activates when CPU usage exceeds 80% use the following commands:

```
               awplus# configure terminal
       awplus(config)# trigger 28
awplus(config-trigger)# type cpu 80 up
```

To configure trigger 5 to be a CPU trigger that activates when CPU usage either rises above or drops below 65%, use the following commands:

```
               awplus# configure terminal
       awplus(config)# trigger 5
awplus(config-trigger)# type cpu 65
```

or

```
               awplus# configure terminal
       awplus(config)# trigger 5
awplus(config-trigger)# type cpu 65 any
```

Related Commands show trigger
trigger

type interface

This command configures a trigger to activate based on the link status of an interface. The trigger can be activated when the interface becomes operational by using the **up** option, or when the interface closes by using the **down** option. The trigger can also be configured to activate when either one of these events occurs by using the **any** option.

Syntax

```
type interface <interface> [up|down|any]
```

Parameter	Description
`<interface>`	Interface name. This can be the name of a switch port, an eth-management port, or a VLAN.
`up`	Activate when interface becomes operational.
`down`	Activate when the interface closes.
`any`	Activate when any interface link status event occurs.

Mode Trigger Configuration

Example To configure trigger `19` to be an interface trigger that activates when `port1.0.2` becomes operational, use the following commands:

```
               awplus# configure terminal
       awplus(config)# trigger 19
awplus(config-trigger)# type interface port1.0.2 up
```

Related Commands show trigger
trigger

type memory

This command configures a trigger to activate based on RAM usage level. Selecting the **up** option causes the trigger to activate when memory usage exceeds the specified level. Selecting the **down** option causes the trigger to activate when memory usage drops below the specified level. Selecting **any** causes the trigger to activate in both situations. The default is **any**.

Syntax

```
type memory <1-100> [up|down|any]
```

Parameter	Description
`<1-100>`	The percentage of memory usage at which to trigger.
`up`	Activate when memory usage exceeds the specified level.
`down`	Activate when memory usage drops below the specified level.
`any`	Activate when memory usage passes the specified level in either direction.

Mode Trigger Configuration

Examples To configure trigger 12 to be a memory trigger that activates when memory usage exceeds 50% use the following commands:

```
           awplus# configure terminal
   awplus(config)# trigger 12
awplus(config-trigger)# type memory 50 up
```

To configure trigger 40 to be a memory trigger that activates when memory usage either rises above or drops below 65%, use the following commands:

```
           awplus# configure terminal
   awplus(config)# trigger 40
awplus(config-trigger)# type memory 65
```

or

```
           awplus# configure terminal
   awplus(config)# trigger 40
awplus(config-trigger)# type memory 65 any
```

Related Commands show trigger
trigger

type periodic

This command configures a trigger to be activated at regular intervals. The time period between activations is specified in minutes.

Syntax

```
type periodic <1-1440>
```

Parameter	Description
<1-1440>	The number of minutes between activations.

Mode Trigger Configuration

Usage A combined limit of 10 triggers of the type periodic and time can be configured. If you attempt to add more than 10 triggers the following error message is displayed:

```
% Cannot configure more than 10 triggers with the type time or
periodic
```

For an example trigger configuration that uses the **type periodic** command, see "See Daily Statistics" on page 71.6.

Example To configure trigger 44 to activate periodically at 10 minute intervals use the following commands:

```
                awplus#  configure terminal
        awplus(config)#  trigger 44
awplus(config-trigger)#  type periodic 10
```

Related Commands show trigger
trigger

type ping-poll

This command configures a trigger that activates when Ping Polling identifies that a target device's status has changed. This allows you to run a configuration script when a device becomes reachable or unreachable.

Syntax

```
type ping-poll <1-100> {up|down}
```

Parameter	Description
<1-100>	The ping poll ID.
up	The trigger activates when ping polling detects that the target is reachable.
down	The trigger activates when ping polling detects that the target is unreachable.

Mode Trigger Configuration

Example To configure trigger 106 to activate when ping poll 12 detects that its target device is now unreachable, use the following commands:

```
               awplus# configure terminal
       awplus(config)# trigger 106
awplus(config-trigger)# type ping-poll 12 down
```

Related Commands show trigger
trigger

type reboot

This command configures a trigger that activates when your device is rebooted.

Syntax

```
type reboot
```

Mode Trigger Configuration

Example To configure trigger 32 to activate when your device reboots, use the following commands:

```
          awplus# configure terminal
    awplus(config)# trigger 32
awplus(config-trigger)# type reboot
```

Related Commands

show trigger
trigger

type stack disabled-master

This command (configured to the stack) configures a trigger to activate on a stack member if it becomes the disabled master.

A disabled master has the same configuration as the active master, but has all its links shutdown.

Although this command could activate any trigger script, the intention here is that the script will reactivate the links from their previously shutdown state, to enable the user to manage the switch. An appropriate trigger script must already exist that will apply the no shutdown command on page 12.13 on the deactivated links.

Caution **It is important that any ports that are configured as trunked ports across master and stack members are disabled at their stack member termination when operating in the fallback configuration. Otherwise, the trunked ports will not function correctly on the switch that is connected downstream.**

If the stack virtual-mac command on page 78.38 command is enabled, the stack uses a virtual MAC address. The stack will always use this MAC address and the new elected master will still retain the originally configured virtual MAC address. If the **stack virtual-mac** command is disabled, the stack will use the MAC address of the current master. If the stack master fails, the stack MAC address changes to reflect the new master's MAC address. See "Fixed or Virtual MAC Addressing" on page 77.12 for information on virtual MAC addresses.

Syntax

```
type stack disabled-master
```

Mode Trigger Configuration

Examples To configure trigger 82 to activate on a device if it becomes the disabled master, use the commands:

Command	Description

`awplus#`
`configure terminal`
	Enter the Global Configuration mode

`awplus(config)#`
`trigger 82`
	Enter the Trigger Configuration mode for trigger 82

`awplus(config-trigger)#`
`type stack disabled master`
	Sets the type of trigger

`awplus(config-trigger)#`
`script 1 flash:/disabled.scp`	

`awplus(config-trigger)#`
`exit`	

Related Commands

stack disabled-master-monitoring
trigger
type stack master-fail
type stack member
type stack link

type stack master-fail

This command (configured to the stack) initiates the action of a pre-configured trigger to occur when the stack enters the fail-over state.

Syntax

```
type stack master-fail
```

Mode Trigger Configuration

Example To configure trigger 86 to activate when stack master fail-over event occurs, use the commands:

```
             awplus#  configure terminal
     awplus(config)#  trigger 86
awplus(config-trigger)#  type stack master-fail
```

Related Commands

stack disabled-master-monitoring
trigger
type stack disabled-master
type stack member
type stack link

type stack member

This command (configured to the stack) initiates the action of a pre-configured trigger to occur when a switch either joins or leaves the stack.

Syntax

```
type stack member {join|leave}
```

Parameter	Description
`join`	Neighbor join event
`leave`	Neighbor leave event

Mode Trigger Configuration

Example To configure a pre-configured trigger number 86 to activate when a new switch joins the stack Note that the number 86 has no particular significance; you can assign any (previously created) numbered trigger

```
                awplus# configure terminal
        awplus(config)# trigger 86
awplus(config-trigger)# type stack member join
```

Related Commands trigger
type stack master-fail
type stack link

type stack link

This command (configured to the stack) initiates the action of a pre-configured trigger to occur when a stacking link is either activated or deactivated.

Syntax

```
type stack link {up|down}
```

Parameter	Description
`up`	Stack link up event
`down`	Stack link down event

Mode Trigger Configuration

Example To configure trigger 86 to activate when the stack link down event occurs, use the commands:

```
                awplus# configure terminal
        awplus(config)# trigger 86
awplus(config-trigger)# type stack link down
```

Related Commands

show trigger
trigger
type stack master-fail

type time

This command configures a trigger that activates at a specified time of day.

Syntax

```
type time <hh:mm>
```

Parameter	Description
`<hh:mm>`	The time to activate the trigger.

Mode Trigger Configuration

Usage A combined limit of 10 triggers of the type time and type periodic can be configured. If you attempt to add more than 10 triggers the following error message is displayed:

```
% Cannot configure more than 10 triggers with the type time or
periodic
```

Example To configure trigger 86 to activate at 15:53, use the following commands:

```
             awplus# configure terminal
     awplus(config)# trigger 86
awplus(config-trigger)# type time 15:53
```

Related Commands show trigger
trigger

undebug trigger

This command applies the functionality of the no debug trigger command.

Chapter 73: Ping Polling Introduction and Configuration

Introduction

Ping polling lets your device regularly check whether it can reach other hosts on a network. It works by sending ICMP Echo Requests to a host and waiting for replies sent back. If ping polling indicates that a host's status has changed, then your device can respond to the new status. When a host is unreachable, ping polling continues monitoring the host's reachability.

You can configure triggers to activate when ping polling determines that the host's status has changed. For example, you could configure a trigger to run a script that opens and configures an alternative link if the host at the other end of a preferred link becomes unavailable. You could then configure a second trigger to run a script that automatically returns traffic to the preferred link as soon as it is available again.

How Ping Polling Works

To determine a host's reachability, your device regularly sends ICMP Echo Request packets ("pings") to the host. As long as your device receives ping responses from the host, it considers the host to be reachable. If your device does not receive a reply to a set number of ICMP Echo Requests, it considers that the host is unreachable. It continues to try to ping the device, at an increased rate. After it receives a set number of responses, it considers the device to be reachable again.

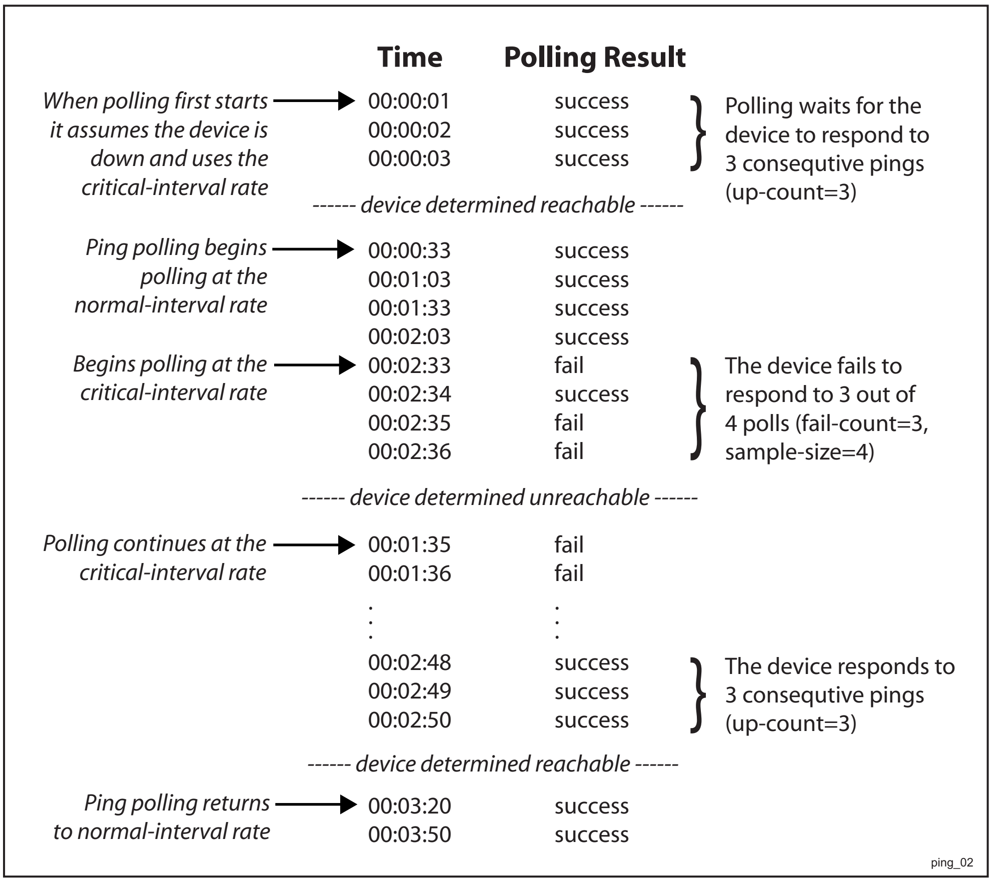
By default, a polling instance sends a ping every 30 seconds as long as it is receiving replies. The frequency of this polling is controlled by the normal-interval command. When a reply is not received, the polling instance increases the frequency at which it polls the device. This frequency is controlled by the critical-interval command, and by default, is set to send a packet every one second. It maintains this higher rate of polling until it has received sufficient consecutive replies.

The polling instance determines whether a device is reachable or unreachable based on the settings of the fail-count, sample-size, and up-count commands. To determine whether a device is reachable, the polling instance counts the number of failed pings within a set sample size. The sample size is set by the sample-size command, and by default is 5 ping responses. Within the sample size, the number of failed pings that means that the device is down is set by the fail-count command. By default this is set to 5. Once a polling instance has determined that a device is unreachable, it must receive a set number of consecutive replies before it changes the device's status back to reachable. This number is configured with the up-count command.

The following figure illustrates a polling instance where the device becomes unreachable, then reachable. It uses this configuration:

```
awplus(config-ping-poll)# fail-count 4

awplus(config-ping-poll)# sample-size 5

awplus(config-ping-poll)# up-count 3

awplus(config-ping-poll)# critical-interval 1

awplus(config-ping-poll)# normal-interval 30
```

Figure 73-1: Interaction between states and parameters for ping polling


Time
Polling Result
When polling first starts it assumes the device is down and uses the critical-interval rate
00:00:01 success
00:00:02 success
00:00:03 success
Polling waits for the device to respond to 3 consequtive pings (up-count=3)
------ device determined reachable ------
Ping polling begins polling at the normal-interval rate
00:00:33 success
00:01:03 success
00:01:33 success
00:02:03 success
Begins polling at the critical-interval rate
00:02:33 fail
00:02:34 success
00:02:35 fail
00:02:36 fail
The device fails to respond to 3 out of 4 polls (fail-count=3, sample-size=4)
------ device determined unreachable ------
Polling continues at the critical-interval rate
00:01:35 fail
00:01:36 fail
00:02:48 success
00:02:49 success
00:02:50 success
The device responds to 3 consequtive pings (up-count=3)
------ device determined reachable ------
Ping polling returns to normal-interval rate
00:03:20 success
00:03:50 success
ping_02

On some operating systems, some servers may respond to a ping even if no other functionality is available, and therefore remain in an Up state while malfunctioning.

Responding to status changes

To configuring your device to determine and respond to changes in a device's reachability, you will need to:

- create a polling instance to periodically ping the device
- create scripts to run when the device becomes unreachable and when it becomes reachable again
- configure triggers to run these scripts

To set a trigger to activate when a device's status changes, its trigger type must be **ping-poll**. This is with the following command in the trigger's configuration mode:

```
awplus(config-trigger)# type ping-poll <1-100> {up|down}
```

where **up** activates the trigger when the device is reachable, and **down** activates the trigger when the device is unreachable.

If you use triggers to open a backup link to a remote device in the event of the primary link failing (rather than the remote device failing), the backup link and primary link must point to different IP addresses on the remote device. Otherwise, when the backup link points to the IP address that your device is polling, your device receives ping replies through the backup link, considers the device to be reachable again, and attempts to reopen the primary link instead of using the backup link. See Chapter 70, Triggers Introduction for more information about configuring Triggers with Ping Polling.

Configuring Ping Polling

This section contains:

- Creating a Polling Instance

  This explains how to quickly create a polling instance using the ping polling defaults.
- Customizing a Polling Instance

  This explains how to customize a ping poll and explains the other ping poll commands.
- Troubleshooting Ping Polling

  This explains how to use the debugging and monitoring commands for ping polling.

Creating a Polling Instance

The Ping Polling feature in the AlliedWare Plus^{TM} OS allows you to easily configure polling instances with a minimum of commands. To configure a ping poll suitable for most network situations:

1. Create a polling instance by using the command:

   ```
   awplus(config)# ping-poll <1-100>
   ```

   The range **<1-100>** identifies the polling instance in the trigger commands and in other ping poll commands. Your device can poll up to 100 IP addresses at once.

2. Set the IP address of the device you are polling by using the command:

   ```
   awplus(config-ping-poll)# ip {<ip-address>|<ipv6-address>}
   ```

3. Enable the polling instance by using the command:

   ```
   awplus(config-ping-poll)# active
   ```

4. If desired, set an optional description to identify the polling instance, by using the command:

   ```
   awplus(config-ping-poll)# description <description>
   ```

You do not need to configure any other commands for most networks, because convenient defaults exist for all other ping poll settings. The following table summarizes the default configuration created.

Command	Default
Critical-interval	1 second
Fail-count	5
Length	32 bytes
Normal-interval	30 seconds
Sample-size	5

Command(cont.)	Default(cont.)
Source-ip	The IP address of the interface from which the ping packets are transmitted
Time-out	1 second
Up-count	30

Customizing a Polling Instance

Once you've created a polling instance using the ping-poll and ip (ping-polling) command, you may wish to customize the polling instance for your network.

Packet size

If you find that larger packet types in your network are not reaching the polled device while smaller ones such as ping do, you can increase the data bytes included in the ping packets sent by the polling instance. This encourages the polling instance to change the device's status to unreachable when packet of the size you are interested in are being dropped. To change the number of bytes sent in the data portion of the ping packets, use the command:

```
awplus(config-ping-poll)# length <4-1500>
```

Response timeout

The polling instance determines that a device hasn't responded to a ping if one second elapses without a response to the ping. In networks where ping packets have a low priority, you may need to set the allowed response time to a longer time period. To change this, use the command:

```
awplus(config-ping-poll)# timeout <1-30>
```

Polling frequency

By default, a polling instance polls a reachable device every 30 seconds. You can change this by using the command:

```
awplus(config-ping-poll)# normal-interval <1-65536>
```

Once the polling instance has determined that a ping has failed, it starts polling the device at the frequency set as the critical interval—by default, one second. To change the frequency set by the critical interval, use the command:

```
awplus(config-ping-poll)# critical-interval <1-65536>
```

The critical interval enables the polling instance to quickly observe changes in the state of the device, and should be set to a much lower value than the normal interval.

Configuring when the device's status changes

The number of pings that the polling instance examines to consider a change in state is controlled by the interaction of the sample-size, fail-count, and up-count commands. See "How Ping Polling Works" on page 73.2 for an example showing this interaction.

To determine whether a device is reachable, the polling instance counts the number of failed pings within a sample of a set size. The sample size is 5 pings by default. To change the sample size, use the command:

```
awplus(config-ping-poll)# sample-size <1-100>
```

To change the number of failed pings that the sample must have, use the command:

```
awplus(config-ping-poll)# fail-count <1-100>
```

If the sample size and fail count are the same, the unanswered pings must be consecutive. If the sample size is greater than the fail count, a device that does not always reply to pings may be declared unreachable.

The upcount is the number of consecutive pings that must be answered for the polling instance to consider the device reachable again. To change this from the default of 30, use the command:

```
awplus(config-ping-poll)# up-count <1-100>
```

Checking the configuration

To check the settings and status of the polling instance, use the command:

```
awplus(config-ping-poll)# show ping-poll [<1-100>|state {up|
                          down}] [brief]
```

Troubleshooting Ping Polling

To disable a polling instance, use the command:

```
awplus(config-ping-poll)# no active
```

The polling instance no longer sends ICMP echo requests to the polled device and the counters for this polling instance are reset.

To clear the counters and change the status of a device to unreachable, enter the Privileged Exec mode and use the command:

```
awplus# clear ping-poll {<1-100>|all}
```

The polling instance changes to the polling frequency specified with the **critical-interval** command. The device status changes to reachable once the device responses have reached the **up-count**.

To start debugging for ping polling, use the command:

```
awplus# debug ping-poll <1-100>
```

Interaction with Other Protocols

Ping polling does not work if the polled host, your device, or any intermediate routers or switches are configured to drop ICMP Echo Requests and Replies.

Ping and Traceroute

Ping and Traceroute are not affected by ping polling. You can enter ping and trace commands at any time and independent of the polling.

Chapter 74: Ping-Polling Commands

Command List

This chapter provides an alphabetical reference for commands used to configure Ping Polling. For more information, see Chapter 73, Ping Polling Introduction and Configuration.

For information about modifying or redirecting the output from **show** commands to a file, see "Controlling "show" Command Output" on page 1.41.

Table 74-1: The following table lists the default values when configuring a ping poll.

Default	Value
Critical-interval	1 second
Description	No description
Fail-count	5
Length	32 bytes
Normal-interval	30 seconds
Sample-size	5
Source-ip	The IP address of the interface from which the ping packets are transmitted
Time-out	1 second
Up-count	30

active (ping-polling)

This command enables a ping-poll instance. The polling instance sends ICMP echo requests to the device with the IP address specified by the ip (ping-polling) command.

By default, polling instances are disabled. When a polling instance is enabled, it assumes that the device it is polling is unreachable.

The **no** variant of this command disables a ping-poll instance. The polling instance no longer sends ICMP echo requests to the polled device. This also resets all counters for this polling instance.

Syntax

```
active

no active
```

Mode Ping-Polling Configuration

Examples To activate the ping-poll instance 43, use the commands:

```
awplus# configure terminal
awplus(config)# ping-poll 43
awplus(config-ping-poll)# active
```

To disable the ping-poll instance 43 and reset its counters, use the commands:

```
awplus# configure terminal
awplus(config)# ping-poll 43
awplus(config-ping-poll)# no active
```

Related Commands

debug ping-poll
ip (ping-polling)
ping-poll
show ping-poll

clear ping-poll

This command resets the specified ping poll, or all ping poll instances. This clears the ping counters, and changes the status of polled devices to unreachable. The polling instance changes to the polling frequency specified with the critical-interval command. The device status changes to reachable once the device responses have reached the up-count.

Syntax

```
clear ping-poll {<1-100>|all}
```

Parameter	Description
`<1-100>`	A ping poll ID number. The specified ping poll instance has its counters cleared, and the status of the device it polls is changed to unreachable.
`all`	Clears the counters and changes the device status of all polling instances.

Mode Privileged Exec

Examples To reset the ping poll instance 12, use the command:

```
awplus# clear ping-poll 12
```

To reset all ping poll instances, use the command:

```
awplus# clear ping-poll all
```

Related Commands

active (ping-polling)
ping-poll
show ping-poll

critical-interval

This command specifies the time period in seconds between pings when the polling instance has not received a reply to at least one ping, and when the device is unreachable.

This command enables the device to quickly observe changes in state, and should be set to a much lower value than the normal-interval command.

The **no** variant of this command sets the critical interval to the default of one second.

Syntax

```
critical-interval <1-65536>
```

```
no critical-interval
```

Parameter	Description
<1-65536>	Time in seconds between pings, when the device has failed to a ping, or the device is unreachable.

Default The default is 1 second.

Mode Ping-Polling Configuration

Examples To set the critical interval to 2 seconds for the ping-polling instance 99, use the commands:

```
awplus# configure terminal
awplus(config)# ping-poll 99
awplus(config-ping-poll)# critical-interval 2
```

To reset the critical interval to the default of one second for the ping-polling instance 99, use the commands:

```
awplus# configure terminal
awplus(config)# ping-poll 99
awplus(config-ping-poll)# no critical-interval
```

Related Commands

fail-count
normal-interval
sample-size
show ping-poll
timeout (ping polling)
up-count

debug ping-poll

This command enables ping poll debugging for the specified ping-poll instance. This generates detailed messages about ping execution.

The **no** variant of this command disables ping-poll debugging for the specified ping-poll.

Syntax

```
debug ping-poll <1-100>

no debug ping-poll {<1-100>|all}
```

Parameter	Description
`<1-100>`	A unique ping poll ID number.
`all`	Turn off all ping-poll debugging.

Mode Privileged Exec

Examples To enable debugging for ping-poll instance 88, use the command:

```
awplus# debug ping-poll 88
```

To disable all ping poll debugging, use the command:

```
awplus# no debug ping-poll all
```

To disable debugging for ping-poll instance 88, use the command:

```
awplus# no debug ping-poll 88
```

Related Commands

active (ping-polling)
clear ping-poll
ping-poll
show ping-poll
undebug ping-poll

description (ping-polling)

This command specifies a string to describe the ping-polling instance. This allows the ping-polling instance to be recognized easily in show commands. Setting this command is optional.

By default ping-poll instances do not have a description.

Use the **no** variant of this command to delete the description set.

Syntax

```
description <description>

no description
```

Parameter	Description
<description>	The description of the target. Valid characters are any printable character and spaces. There is no maximum character length.

Mode Ping-Polling Configuration

Examples To add the text “Primary Gateway” to describe the ping-poll instance 45, use the commands:

```
awplus# configure terminal
awplus(config)# ping-poll 45
awplus(config-ping-poll)# description Primary Gateway
```

To delete the description set for the ping-poll instance 45, use the commands:

```
awplus# configure terminal
awplus(config)# ping-poll 45
awplus(config-ping-poll)# no description
```

Related Commands ping-poll
show ping-poll

fail-count

This command specifies the number of pings that must be unanswered, within the total number of pings specified by the sample-size command, for the ping-polling instance to consider the device unreachable.

If the number set by the sample-size command and the **fail-count** commands are the same, then the unanswered pings must be consecutive. If the number set by the sample-size command is greater than the number set by the **fail-count** command, then a device that does not always reply to pings may be declared unreachable.

The **no** variant of this command resets the fail count to the default.

Syntax

```
fail-count <1-100>
```

```
no fail-count
```

Parameter	Description
<1-100>	The number of pings within the sample size that a reachable device must fail to respond to before it is classified as unreachable.

Default The default is 5.

Mode Ping-Polling Configuration

Examples To specify the number of pings that must fail within the sample size to determine that a device is unreachable for ping-polling instance 45, use the commands:

```
awplus# configure terminal
awplus(config)# ping-poll 45
awplus(config-ping-poll)# fail-count 5
```

To reset the fail-count to its default of 5 for ping-polling instance 45, use the commands:

```
awplus# configure terminal
awplus(config)# ping-poll 45
awplus(config-ping-poll)# no fail-count
```

Related Commands

critical-interval
normal-interval
ping-poll
sample-size
show ping-poll
timeout (ping polling)
up-count

ip (ping-polling)

This command specifies the IPv4 address of the device you are polling.

Syntax `ip {<ip-address>|<ipv6-address>}`

Parameter	Description
<ip-address>	An IPv4 address in dotted decimal notation `A.B.C.D`
<ipv6-address>	An IPv6 address in hexadecimal notation `X:X::X:X`

Mode Ping-Polling Configuration

Examples To set ping-poll instance 5 to poll the device with the IP address `192.168.0.1`, use the commands:

```
awplus# configure terminal
awplus(config)# ping-poll 5
awplus(config-ping-poll)# ip 192.168.0.1
```

To set ping-poll instance 10 to poll the device with the IPv6 address `2001:db8::`, use the commands:

```
awplus# configure terminal
awplus(config)# ping-poll 10
awplus(config-ping-poll)# ip 2001:db8::
```

Related Commands ping-poll
source-ip
show ping-poll

length (ping-poll data)

This command specifies the number of data bytes to include in the data portion of the ping packet. This allows you to set the ping packets to a larger size if you find that larger packet types in your network are not reaching the polled device, while smaller packets are getting through. This encourages the polling instance to change the device's status to unreachable when the network is dropping packets of the size you are interested in.

The **no** variant of this command resets the data bytes to the default of 32 bytes.

Syntax

```
length <4-1500>

no length
```

Parameter	Description
<4-1500>	The number of data bytes to include in the data portion of the ping packet.

Default The default is 32.

Mode Ping-Polling Configuration

Examples To specify that ping-poll instance 12 sends ping packet with a data portion of 56 bytes, use the commands:

```
awplus# configure terminal
awplus(config)# ping-poll 12
awplus(config-ping-poll)# length 56
```

To reset the number of data bytes in the ping packet to the default of 32 bytes for ping-poll instance 3, use the commands:

```
awplus# configure terminal
awplus(config)# ping-poll 3
awplus(config-ping-poll)# ping-poll 3
```

Related Commands ping-poll
show ping-poll

normal-interval

This command specifies the time period between pings when the device is reachable.

The **no** variant of this command resets the time period to the default of 30 seconds.

Syntax

```
normal-interval <1-65536>
```

```
no normal-interval
```

Parameter	Description
<1-65536>	Time in seconds between pings when the target is reachable.

Default The default is 30 seconds.

Mode Ping-Polling Configuration

Examples To specify a time period of 60 seconds between pings when the device is reachable for ping-poll instance 45, use the commands:

```
awplus# configure terminal
awplus(config)# ping-poll 45
awplus(config-ping-poll)# normal-interval 60
```

To reset the interval to the default of 30 seconds for ping-poll instance 45, use the commands:

```
awplus# configure terminal
awplus(config)# ping-poll 45
awplus(config-ping-poll)# no normal-interval
```

Related Commands

critical-interval
fail-count
ping-poll
sample-size
show ping-poll
timeout (ping polling)
up-count

ping-poll

This command enters the ping-poll configuration mode. If a ping-poll exists with the specified number, then this command enters its configuration mode. If no-ping poll exists with the specified number, then this command creates a new ping poll with this ID number.

To configure a ping-poll, create a ping poll using this command, and use the ip (ping-polling) command to specify the device you want the polling instance to poll. It is not necessary to specify any further commands unless you want to change a command's default.

The **no** variant of this command deletes the specified ping poll.

Syntax

```
ping-poll <1-100>

no ping-poll <1-100>
```

Parameter	Description
<1-100>	A unique ping poll ID number.

Mode Global Configuration

Examples To create ping-poll instance 3 and enter ping-poll configuration mode, use the commands:

```
awplus# configure terminal

awplus(config)# ping-poll 3

awplus(config-ping-poll)#
```

To delete ping-poll instance 3, use the commands:

```
awplus# configure terminal

awplus(config)# no ping-poll 3
```

Related Commands

active (ping-polling)
clear ping-poll
debug ping-poll
description (ping-polling)
ip (ping-polling)
length (ping-poll data)
show ping-poll
source-ip

sample-size

This command sets the total number of pings that the polling instance inspects when determining whether a device is unreachable. If the number of pings specified by the **fail-count** command go unanswered within the inspected sample, then the device is declared unreachable.

If the numbers set in this command and fail-count command are the same, the unanswered pings must be consecutive. If the number set by this command is greater than that set with the fail-count command, a device that does not always reply to pings may be declared unreachable.

You cannot set this command's value lower than the fail-count value.

The polling instance uses the number of pings specified by the up-count command to determine when a device is reachable.

The **no** variant of this command resets this command to the default.

Syntax

```
sample-size <1-100>
```

```
no sample size
```

Parameter	Description
<1-100>	Number of pings that determines critical and up counts.

Default The default is 5.

Mode Ping-Polling Configuration

Examples To set the sample-size to 50 for ping-poll instance 43, use the commands:

```
awplus# configure terminal
```

```
awplus(config)# ping-poll 43
```

```
awplus(config-ping-poll)# sample-size 50
```

To reset sample-size to the default of 5 for ping-poll instance 43, use the commands:

```
awplus# configure terminal
```

```
awplus(config)# ping-poll 43
```

```
awplus(config-ping-poll)# no sample-size
```

Related Commands

critical-interval
fail-count
normal-interval
ping-poll
show ping-poll
timeout (ping polling)
up-count

show counter ping-poll

This command displays the counters for ping polling.

Syntax `show counter ping-poll [<1-100>]`

Parameter	Description
<1-100>	A unique ping poll ID number. This displays the counters for the specified ping poll only. If you do not specify a ping poll, then this command displays counters for all ping polls.

Mode User Exec and Privileged Exec

Output Figure 74-1: Example output from the **show counter ping-poll** command

```
Ping-polling counters
Ping-poll: 1
PingsSent              ......... 15
PingsFailedUpState     ......... 0
PingsFailedDownState   ......... 0
ErrorSendingPing       ......... 2
CurrentUpCount         ......... 13
CurrentFailCount       ......... 0
UpStateEntered         ......... 0
DownStateEntered       ......... 0

Ping-poll: 2
PingsSent              ......... 15
PingsFailedUpState     ......... 0
PingsFailedDownState   ......... 0
ErrorSendingPing       ......... 2
CurrentUpCount         ......... 13
CurrentFailCount       ......... 0
UpStateEntered         ......... 0
DownStateEntered       ......... 0

Ping-poll: 5
PingsSent              ......... 13
PingsFailedUpState     ......... 0
PingsFailedDownState   ......... 2
ErrorSendingPing       ......... 2
CurrentUpCount         ......... 9
CurrentFailCount       ......... 0
UpStateEntered         ......... 0
DownStateEntered       ......... 0
```

Table 74-2: Parameters in output of the **show counter ping-poll** command

Parameter	Description
`Ping-poll`	The ID number of the polling instance.
`PingsSent`	The total number of pings generated by the polling instance.
`PingsFailedUpState`	The number of unanswered pings while the target device is in the Up state. This is a cumulative counter for multiple occurrences of the Up state.
`PingsFailedDownState`	Number of unanswered pings while the target device is in the Down state. This is a cumulative counter for multiple occurrences of the Down state.

Table 74-2: Parameters in output of the **show counter ping-poll** command(cont.)

Parameter	Description
`ErrorSendingPing`	The number of pings that were not successfully sent to the target device. This error can occur when your device does not have a route to the destination.
`CurrentUpCount`	The current number of sequential ping replies.
`CurrentFailCount`	The number of ping requests that have not received a ping reply in the current sample-size window.
`UpStateEntered`	Number of times the target device has entered the Up state.
`DownStateEntered`	Number of times the target device has entered the Down state.

Example To display counters for the polling instances, use the command:

```
awplus# show counter ping-poll
```

Related Commands debug ping-poll
ping-poll
show ping-poll

show ping-poll

This command displays the settings and status of ping polls.

Syntax `show ping-poll [<1-100>|state {up|down}] [brief]`

Parameter	Description	
<1-100>	Displays settings and status for the specified polling instance.	
`state`	Displays polling instances based on whether the device they are polling is currently reachable or unreachable.	
	`up`	Displays polling instance where the device state is reachable.
	`down`	Displays polling instances where the device state is unreachable.
`brief`	Displays a summary of the state of ping polls, and the devices they are polling.	

Mode User Exec and Privileged Exec

Output Figure 74-2: Example output from the **show ping-poll brief** command

```
Ping Poll Configuration
----------------------------------------------------------
Id Enabled State Destination
----------------------------------------------------------
1  Yes     Down  192.168.0.1
2  Yes     Up    192.168.0.100
```

Table 74-3: Parameters in output of the show ping-poll brief command

Parameter	Meaning	
Id	The ID number of the polling instance, set when creating the polling instance with the ping-poll command.	
Enabled	Whether the polling instance is enabled or disabled.	
`State`	The current status of the device being polled:	
	`Up`	The device is reachable.
	`Down`	The device is unreachable.
	`Critical Up`	The device is reachable but recently the polling instance has not received some ping replies, so the polled device may be going down.
	`Critical Down`	The device is unreachable but the polling instance received a reply to the last ping packet, so the polled device may be coming back up.

Table 74-3: Parameters in output of the show ping-poll brief command(cont.)

Parameter	Meaning
Destination	The IP address of the polled device, set with the ip (ping-polling) command.

Figure 74-3: Example output from the show ping-poll command

```
Ping Poll Configuration
----------------------------------------------------------

Poll 1:
Description                         : Primary Gateway
Destination IP address              : 192.168.0.1
Status                              : Down
Enabled                             : Yes
Source IP address                   : 192.168.0.10
Critical interval                   : 1
Normal interval                     : 30
Fail count                          : 10
Up count                            : 5
Sample size                         : 50
Length                              : 32
Timeout                             : 1
Debugging                           : Enabled


Poll 2:
Description                         : Secondary Gateway
Destination IP address              : 192.168.0.100
Status                              : Up
Enabled                             : Yes
Source IP address                   : Default
Critical interval                   : 5
Normal interval                     : 60
Fail count                          : 20
Up count                            : 30
Sample size                         : 100
Length                              : 56
Timeout                             : 2
Debugging                           : Enabled
```

Table 74-4: Parameters in output of the show ping-poll command

Parameter	Description
Description	Optional description set for the polling instance with the description (ping-polling) command.
Destination IP address	The IP address of the polled device, set with the ip (ping-polling) command.

Table 74-4: Parameters in output of the show ping-poll command(cont.)

Parameter	Description	
`Status`	The current status of the device being polled:	
	`Up`	The device is reachable.
	`Down`	The device is unreachable.
	`Critical Up`	The device is reachable but recently the polling instance has not received some ping replies, so the polled device may be going down.
	`Critical Down`	The device is unreachable but the polling instance received a reply to the last ping packet, so the polled device may be coming back up.
`Enabled`	Whether the polling instance is enabled or disabled. The active (ping-polling) and no active commands enable and disable a polling instance.	
`Source IP address`	The source IP address sent in the ping packets. This is set using the source-ip command.	
`Critical interval`	The time period in seconds between pings when the polling instance has not received a reply to at least one ping, and when the device is unreachable. This is set with the critical-interval command.	
`Normal interval`	The time period between pings when the device is reachable. This is set with the normal-interval command.	
`Fail count`	The number of pings that must be unanswered, within the total number of pings specified by the sample-size command, for the polling instance to consider the device unreachable. This is set using the fail-count command.	
`Up count`	The number of consecutive pings that the polling instance must receive a reply to before classifying the device reachable again. This is set using the up-count command.	
`Sample size`	The total number of pings that the polling instance inspects when determining whether a device is unreachable. This is set using the sample-size command.	
`Length`	The number of data bytes to include in the data portion of the ping packet. This is set using the length (ping-poll data) command.	
`Timeout`	The time in seconds that the polling instance waits for a response to a ping packet. This is set using the timeout (ping polling) command.	
`Debugging`	Indicates whether ping polling debugging is **Enabled** or **Disabled**. This is set using the debug ping-poll command.	

Examples To display the ping poll settings and the status of all the polls, use the command:

```
awplus# show ping-poll
```

To display a summary of the ping poll settings, use the command:

```
awplus# show ping-poll brief
```

To display the settings for ping poll 6, use the command:

```
awplus# show ping-poll 6
```

To display a summary of the state of ping poll 6, use the command:

```
awplus# show ping-poll 6 brief
```

To display the settings of ping polls that have reachable devices, use the command:

```
awplus# show ping-poll state up
```

To display a summary of ping polls that have unreachable devices, use the command:

```
awplus# show ping-poll 6 state down brief
```

Related Commands

debug ping-poll
ping-poll

source-ip

This command specifies the source IP address to use in ping packets.

By default, the polling instance uses the address of the interface through which it transmits the ping packets. It uses the device's local interface IP address when it is set. Otherwise, the IP address of the interface through which it transmits the ping packets is used.

The **no** variant of this command resets the source IP in the packets to the device's local interface IP address.

Syntax

```
source-ip {<ip-address>|<ip-address>}
```

```
no source-ip
```

Parameter	Description
<ip-address>	An IPv4 address in dotted decimal notation `A.B.C.D`
<ipv6-address>	An IPv6 address in hexadecimal notation `X:X::X:X`

Mode Ping-Polling Configuration

Examples To configure the ping-polling instance 43 to use the source IP address `192.168.0.1` in ping packets, use the commands:

```
awplus# configure terminal
awplus(config)# ping-poll 43
awplus(config-ping-poll)# source-ip 192.168.0.1
```

To configure the ping-polling instance 43 to use the source IPv6 address `2001:db8::` in ping packets, use the commands:

```
awplus# configure terminal
awplus(config)# ping-poll 43
awplus(config-ping-poll)# source-ip 2001:db8::
```

To reset the source IP address to the device's local interface IP address for ping-poll instance 43, use the commands:

```
awplus# configure terminal
awplus(config)# ping-poll 43
awplus(config-ping-poll)# no source-ip
```

Related Commands

description (ping-polling)
ip (ping-polling)
length (ping-poll data)
ping-poll
show ping-poll

timeout (ping polling)

This command specifies the time in seconds that the polling instance waits for a response to a ping packet. You may find a higher time-out useful in networks where ping packets have a low priority.

The **no** variant of this command resets the set time out to the default of one second.

Syntax

```
timeout <1-30>
```

```
no timeout
```

Parameter	Description
<1-30>	Length of time, in seconds, that the polling instance waits for a response from the polled device.

Default The default is 1 second.

Mode Ping-Polling Configuration

Examples To specify the timeout as 5 seconds for ping-poll instance 43, use the commands:

```
awplus# configure terminal
awplus(config)# ping-poll 43
awplus(config-ping-poll)# timeout 5
```

To reset the timeout to its default of 1 second for ping-poll instance 43, use the commands:

```
awplus# configure terminal
awplus(config)# ping-poll 43
awplus(config-ping-poll)# no timeout
```

Related Commands

critical-interval
fail-count
normal-interval
ping-poll
sample-size
show ping-poll
up-count

up-count

This command sets the number of consecutive pings that the polling instance must receive a reply to before classifying the device reachable again.

The **no** variant of this command resets the up count to the default of 30.

Syntax

```
up-count <1-100>
```

```
no up-count
```

Parameter	Description
<1-100>	Number of replied pings before an unreachable device is classified as reachable.

Default The default is 30.

Mode Ping-Polling Configuration

Examples To set the upcount to 5 consecutive pings for ping-polling instance 45, use the commands:

```
awplus# configure terminal
awplus(config)# ping-poll 45
awplus(config-ping-poll)# up-count 5
```

To reset the upcount to the default value of 30 consecutive pings for ping-polling instance 45, use the commands:

```
awplus# configure terminal
awplus(config)# ping-poll 45
awplus(config-ping-poll)# no up-count
```

Related Commands

critical-interval
fail-count
normal-interval
ping-poll
sample-size
show ping-poll
timeout (ping polling)

undebug ping-poll

This command applies the functionality of the no debug ping-poll command on page 74.6.

Chapter 75: sFlow Introduction and Configuration

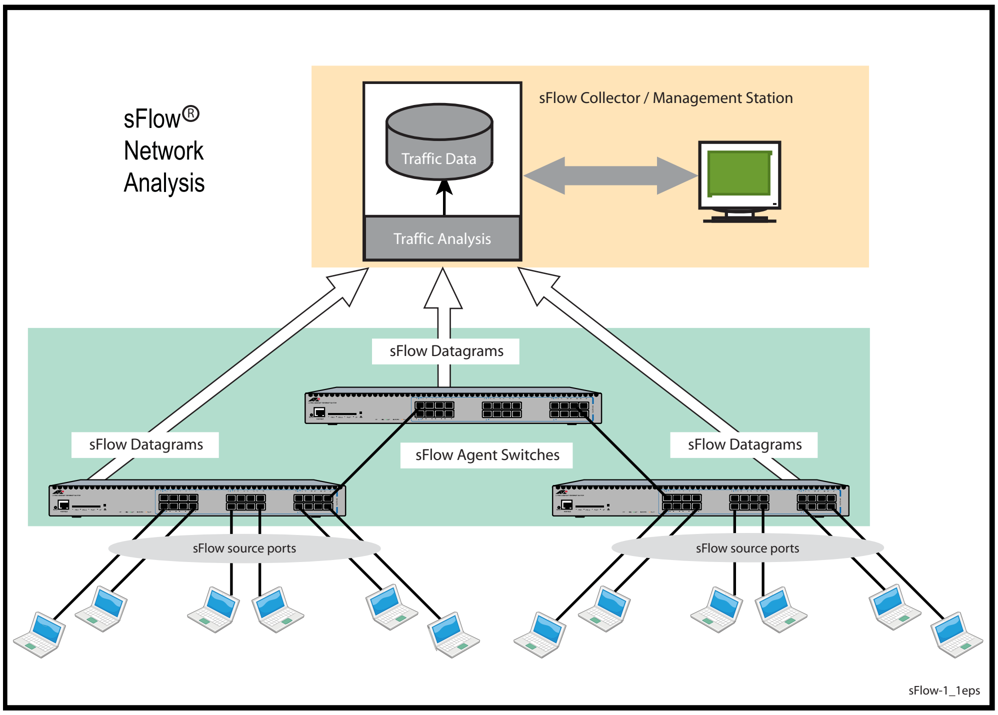
sFlow Introduction

sFlow®[1] provides the ability to monitor traffic in data networks containing switches and routers. A network employing sFlow typically comprises a number of network (sFlow) agents that accumulate sampled data and traffic counter information. The agents then forward this data to a collector. The collector then analyses the information supplied by its agents in order to compile and display statistical profiles of the network and its traffic. The sFlow feature on your switch provides the sFlow Agent capability.

[1] sFlow® is a registered trademark belonging to InMon Corp, San Francisco, CA.

Figure 75-1 on page 75.2 shows a basic sFlow network structure. The three network switches also function as sFlow agents. Each agent switch captures samples of the traffic passing through its monitored ports, and sends these samples together with counter information back to the sFlow collector. The agents sample data from a number of switch ports, each acting as an sFlow data source.

Figure 75-1: Basic sFlow Network



The sFlow Agent

Your switch can act as an sFlow agent. The key capabilities of the agent are to:

- sample frames as they pass through selected ports on the switch, and provide sampled extracts of the network traffic.
- periodically capture interface counter data.
- package together the sampled frame and counter information that can be sent to the collector for analysis and display.
- be configurable via SNMP MIB objects.
- communicate to heterogeneous collector devices by means of standard protocols.

Agent components and functionality

sFlow functionality on your switch is based on the requirements defined in of RFC 3176 and its updates defined in the sFlow version 5 memo dated July 2004. This memo can be found at the web site, www.sflow.org/sflow_version_5.txt.

The terms defined in Table 75-1 are used to describe the agent and its functionality on your switch:

Table 75-1: sFlow Terminology

sFlow® Component	Definition
Network Device	Typically either a network switch or router that has the ability to forward frames across an Ethernet network; or between Ethernet networks, in the case of a router.
Data Source (sFlow Source Port)	The location of a sampling point within the switch. This is typically a switch port.
Packet Flow	The path taken by the data (frames) as they traverse a network device.
Sampling Rate	The ratio of frames passing through the data source, to those captured and forwarded as sFlow data. See **sflow sampling-rate** command on page 76.14.
Counter Sampling	The periodic polling of counters taken at the data source.
sFlow Datagram	A UDP datagram that contains details of sFlow captured data, and counters sent by the sFlow Agent to its Collector.

The sFlow agent (switch) uses sampling technology to derive traffic statistics from its monitored ports. Samples are taken at the sFlow source ports. After collecting its information, the switch then packetizes its samples and statistical data, and sends both to a remote sFlow collector.

Sampling Methods

Two sampling methods are employed within the sFlow agent, frame sampling, and counter sampling. Both sample types are combined within the datagrams sent to the collector. The frame sample data will result in a relatively constant traffic stream, but the counter information is sent where it can fill available space within each datagram. Datagrams are normally sent to the collector at the rate of one each second. However, several datagrams can be sent in rapid succession, where more information exists than can be sent in a single datagram.

Frame Sampling

As frames enter or leave an sFlow source port, they are sampled at a rate determined by the sflow sampling-rate command on page 76.14for that particular port.

Sampling occurs every N frames (on average), where N is the rate value set via sflow sampling-rate command. The sampling rate applies to ingress and egress frames independently. For example, a value of 1000, will sample one frame in every 1000 frames received, one in every 1000 frames sent from the specified port.

Caution ⚠ **Setting the sFlow sampling rate to a very low value (frequent sampling) can place a heavy load on the switch's CPU. The severity of this loading will increase with the number of ports configured for sampling, the port speeds, and their data sampling rates.**

Data Confidentiality

Sampling operates by capturing the initial portion of frames (statistically) selected. The portion sampled is set by the sflow max-header-size command on page 76.11, or SNMP. If the **maximum header size** is greater than the actual headers in the sampled frames, then portions of the user data (payload) will also be captured and encapsulated in the datagrams sent to the collector. The amount of user data captured can be minimized by careful selection of the maximum header size.

Counter Polling

The function of counter polling is to provide snapshots of various system counters. This produces a series of data counter sets for each port, which can be independently polled at user defined rates, and sent (once a second or less) to the collector. Allied Telesis switches running AlliedWare Plus software support generic interface counters only. For more information on the data types included in the sampling count, see "sFlow Datagrams" on page 75.13.

The sFlow Collector

The sFlow collector receives traffic samples and counter information from a number of sFlow agents. These samples are received as a series of UDP datagrams. From the data contained within these datagrams, the collector is able to provide statistical and or graphical information of network traffic.

The sFlow agent application on your switch supports only a single collector configuration.

sFlow collectors are proprietary third party products. Your switch, running as an sFlow agent has been designed for interoperability with any sFlow collector that supports the sFlow Version 5 specification, including the inMon sFlow collector.

The sFlow Collector may also contain an SNMP Manager that is able to configure sFlow on its agent switches.

Configuring sFlow on your Switch

This section provides some guidelines for setting up the sFlow® agent on your switch. sFlow can be configured directly on your switch - using the CLI, or it can be configured via an SNMP manager. The SNMP management function can be carried out either by a the sFlow collector, or a separate SNMP manager. The configuration examples in this section are shown using the CLI.

Caution

The sFlow configurations set either by the switch's CLI, or the sFlow collector. Sometimes the collector will overide the sFlow settings that were initially configured by the CLI, in order to apply "its" own default settings.

If you want to apply the sFlow settings set by the CLI, or by an external network management system, then turn off network management at the collector.

We also advise that as part of your sFlow commissioning process, you review your security access procedures relating to sFlow access and its data traffic management.

sFlow configuration can vary greatly with your overall configuration, data profile, and monitoring intensity. Also, many interdependencies existing between parameter settings. For this reason, few firm configuration settings are recommended in this software reference, but instead these parameter relationships are explained and some typical configuration examples are shown.

The default settings on your switch have sFlow turned off for all ports.
The following commands are used to setup and configure sFlow on your switch. These are introduced in the order in which you would logically need to use them.

sFlow Command	Functionality
sflow enable	enables sFlow on your switch (or stack).
sflow max-header-size	sets the maximum sFlow data capture size.
sflow collector max-datagram-size	sets the maximum size for the agent to collector datagrams.
sflow agent (address)	sets the sFlow agent IP address on the switch.
sflow polling-interval	sets the counter polling interval for specified ports.
sflow sampling-rate	sets the mean sampling rate for specified ports.
sflow collector (address)	the sFlow agent's collector IP address and/or UDP port.

Configuration Procedure

The following process sets out a systematic procedure to configure sFlow on your switch:

Information Gathering

sFlow configuration is dependant on your network structure and its data. Start by gathering together the following information.

- Obtain (or determine) the sFlow collector IP address.
- Select an appropriate UDP port for your sFlow datagrams. The recommended value is 6343, and is the default value preconfigured on your switch.
- Select an appropriate IP address for your sFlow agent. We recommend that you use the local IP address of your switch. For more information on local addresses and how to set them up, see the **interface (to configure)** command on page 12.3.
- Assess the sensitivity of the data that your sFlow agent will be sampling.
- Obtain details of the protocols that your sFlow agent will be sampling. If you intend sampling unusual or proprietary protocols, obtain details of their header lengths.
- Calculate the most appropriate max-header-size for your sFlow sampling.
- Select the ports that you want to sample, and their sample rate.
  These two factors vary (not quite) proportionally; so if you double the number of ports and double your sampling rate (i.e. sample half as many frames) then you will “almost” return to your earlier situation. Also note the speeds of the ports you have selected, because - for the same port utilization - the faster the port speed, the greater the load on the CPU.
- Review the speed of the port used to transport the sFlow datagrams to the collector. Unless configured to a specific port, the collector traffic will share the same network port with other traffic.

  The capacity of the collector port should be sufficient to carry the volume of sFlow traffic. This topic is expanded on in the Configuration Examples later in this chapter.

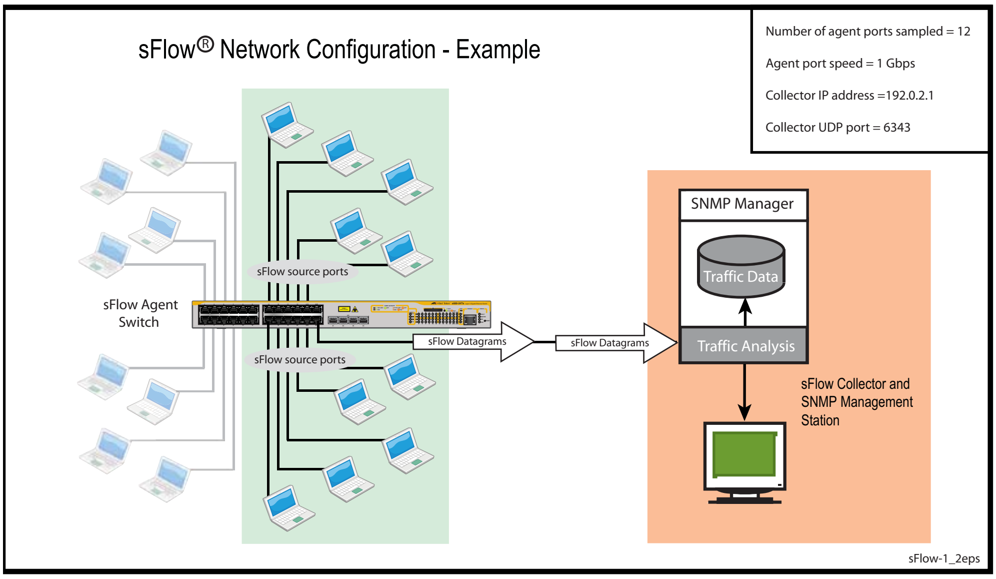
Managing the sFlow processing overhead

The sFlow data sampled on the ports converges into the CPU for processing and UDP packetizing. Therefore one of the major factors when configuring sFlow is to prevent the sFlow data volumes from placing a significant overhead on the CPU processing. The two most significant factors here are, the **number of ports sampled**, and the **sampling rate**. The other (and lessor) factors in this equation are the **frame size distribution** and the **maximum header size.** The shorter the frames are on the network, the heavier the sFlow processing load will be (for the same number of frames per second). Conversely the shorter the maximum header size selected, the lighter the sFlow processing load will be (because less data per frame is sent to the CPU).

Configuration Examples

This configuration example shown is based on the network shown below:

Figure 75-2: sFlow Configuration Example


sFlow® Network Configuration - Example
Number of agent ports sampled = 12
Agent port speed = 1 Gbps
Collector IP address =192.0.2.1
Collector UDP port = 6343
sFlow source ports
sFlow Agent Switch
sFlow Datagrams
sFlow Datagrams
sFlow source ports
SNMP Manager
Traffic Data
Traffic Analysis
sFlow Collector and SNMP Management Station
sFlow-1_2eps

Step 1: Determine the IP addresses and UDP ports

Collector IP address is 192.0.2.65

sFlow UDP port uses the default of 6343

Agent (local) IP address 192.0.2.33. This is the address that the collector may use to configure the agent via SNMP.

Step 2: Determine the maximum sFlow Datagram size

Datagrams will be sent at one second intervals regardless of the amount of data they contain. If the amount of data to be sent is greater than the maximum datagram size, then several datagrams will be sent in quick succession - within the 1 second interval. The objective is to contain the sFlow information in a the minimum number of datagrams. That is. to fragment datagrams when necessary, but do so as little as possible.

Find the maximum datagram size that will pass through all network components without fragmenting. Then set the sFlow datagram size a little less than this value.

The maximum datagram size should be less than the MTU size.

For this example, the MTU is assumed to be set to its default of 1500 bytes. In this situation we could leave the maximum Datagram size at its default of 1400 bytes; but in order to show this as a configuration step, we will change it to 1200 bytes.

Note sFlow datagrams are generally transmitted at 1 second intervals. However, where there is more information than can fit into one datagram, several datagrams are sent sequentially, within the 1 second time frame.

Step 3: Determine the max-header-size sampled data

The maximum header size for the sampled data is set by the **sflow max-header-size** command. The optimum setting is to capture only the header portion of the frame and discard the user-data portion. This is especially important where the user data contains sensitive information.

Keeping the max-header-size as small as possible has the additional benefit of lightening the CPU load.

First, inspect the nature of the data to be sampled and the protocols used to carry it.

For this example we will assume that the network contains Ethernet II frames with the 4 byte 802.1Q header component, IP, TCP protocols. In this situation the following rules can be applied:

For an environment using standard TCP\IPv4 over Ethernet frames, consider the following protocol basics.

Ethernet header (including the 4 byte 802.1Q header component) = 18 bytes

IPv4 header = 24 bytes

TCP header = 24 bytes

Total = 66 bytes

A similar calculation can be made for an environment using IPv6 over Ethernet.

Ethernet header (including the 4 byte 802.1Q component) = 18 bytes

IPv6 header = 40 bytes

TCP header = 24 bytes

Total = 82 bytes

Caution ⚠ In the above network scenarios:

For IPv4—any data existing between 66 bytes and the value set by this command will be included in the sFlow packet samples. For example, with the default of 128 applied, up to 128-66=62 bytes of user data could be included in the sFlow datagram samples sent between the Agent and the Collector.

For IPv6—any data existing between 82 bytes and the value set by this command will be included in the sFlow packet samples. For example, with the default of 128 applied, up to 128-82=46 bytes of user data could be included in the sFlow datagram samples sent between the Agent and the Collector.

For this example the **sflow max-header-size** will be set to 68 bytes (assuming an IPv4 environment)

Step 4: Select ports to sample

Each sampled sFlow port speed is 1 Gbps

12 ports have been selected for sampling

Step 5: Determine the sampling rate

Selecting the sampling rate involves a trade-off between sFlow requirements, and system loading. The greater the sampling rate, the more samples will be taken, and the more accurate their results will be. Unfortunately, taking more samples increases the load on the switch CPU and on the connection to the collector.

For this particular configuration, the value of N was set to 5000 so as to present a light load on the CPU.

Step 6: Review and adjust settings

Because sFlow traffic loading will vary with the traffic profile, the following general assumptions are made. The following traffic profile are assumed.

- 50 % of frames are <200 bytes long
- 40 % of frames are >1400 bytes long

The following settings are:

- 12 x 1 Gbps ports are being sampled
- sFlow max-header size = 68 bytes
- sampling rate (N = 2750)
- average port utilization is assumed to be approximately 60 %
- average data rate to the collector assumed to be approximately 250 Kbps

When setting the sampling rate, consider the following factors that will affect the CPU load. This load will increase (not necessarily linearly) as you:

- increase the number of ports configured
- increase the port speeds
- decrease the sampling rate
- increase the max-header-size

For this configuration the average sFlow collector traffic is expected to be approximately 250 kbps. In this example the agent-to-collector traffic will be shared with non sFlow traffic. Although not described in this example, you can specifically configure the collector port to route only sFlow traffic. To do this you would need to assign a separate VLAN (and IP address) to the agent-to collector interface and direct your sFlow traffic to this interface.

We advise that you ensure adequate bandwidth is provided for both the sFlow and general traffic that could share its network connection.

We will now use these settings to configure the network.

Configuration Procedure

The following steps apply the settings obtained in the previous section.

Step 1: Configure the switch-wide sFlow

Enable sFlow on the switch	
awplus# `configure terminal`	Enter Global Configuration Mode
awplus(config)# `sflow enable`	Enable the sFlow agent globally on the switch

Step 2: Configure the sFlow Collector Settings

Set the sFlow collector max-datagram size	
awplus(config)# `sflow collector max-datagram-size 1200`	Set the maximum size of the sFlow datagrams to 1200 bytes.

Set the sFlow collector (address)	
awplus# `configure terminal`	Enter Global Configuration Mode.
awplus(config)# `sflow collector ip 192.0.2.65`	Set the sFlow collector address to 192.0.2.65

Step 3: Configure the sFlow Agent Settings

Set the sFlow agent (address)	
awplus(config)# `sflow agent ip 192.0.2.33`	Set the sFlow agent address to 192.0.2.33

Set the sFlow sampling rate on sFlow Source Ports	
awplus(config)# `interface port1.0.13-port1.0.22`	Select the port range to configure (ports 1.0.13 to 1.0.22).
awplus(config-if)# `sflow sampling-rate 2750`	Set the sampling rate on the selected ports.

Step 4: Check the Configuration

View the sFlow Configuration	
awplus#	
do show running-config sflow	Validate that sFlow is enabled Note that the prefix "do" enables you to run an Exec Mode command from an Interface Mode prompt.

Figure 75-3: Output from the **show-running config sFlow** command

```
awplus#sh run sflow
!
sflow agent ip 192.0.2.33
sflow collector ip 192.0.2.65
sflow collector max-datagram-size 1200
sflow enable
!
interface port1.0.13-port1.0.22
 sflow sampling-rate 2750
```

sFlow Datagrams

After data sampling and counter information has been gathered, each sFlow agent packetizes the data and sends it to an sFlow collector where it can be analyzed and displayed in charts and tables etc.

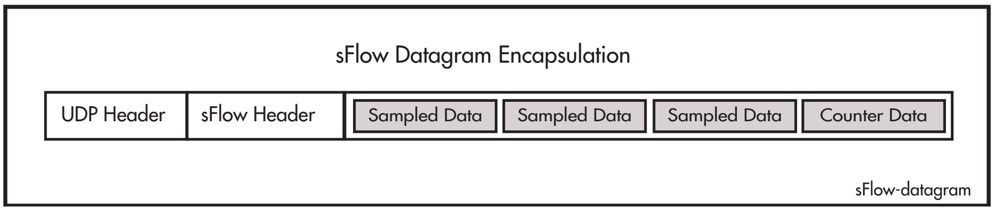
This packetized data is sent to the collector in UDP datagrams. These datagrams bear the IP address of the collector and the port number 6343. Using a standardized port helps to avoid configuration problems between the sFlow agents and collectors.

Although an analysis of the sFlow datagrams is outside the scope of this document, some basic information is provided here for those interested in knowing the basic components of the sFlow datagrams. The full specification of the sFlow protocol can be found at www.sflow.org/sflow_version_5.txt.

sFlow datagrams comprise three basic components:

- Datagram header information
- Flow sample information - may contain several samples
- Counter statistical information - fitted in where space permits

Figure 75-4: sFlow Datagram Encapsulation



The content of these datagram components is listed below:

sFlow Header Fields

- Version (The sFlow version being used)
- IP Address Type (Can be either an IPv4 or IPv6 address type)
- Source IP Address (The IP address of the sFlow agent)
- Sequence Number (The datagram sequence number)
- System Up-time
- Sample Count (The number of samples in the datagram)
- Sample Dataset

sFlow Flow Sample Fields

- Flow Sample 1 (The first sample)
- Sample Type (Flow Sample, 0x0001)
- Sample Sequence Number (of flow samples)
- Sampler ID
- Sampling Rate (as set by the “sflow sampling-rate” on page 76.14 or SNMP)
- Sample Pool (the total number of packets that could have been sampled)
- Packets Dropped (the number of packets dropped, due to a lack of resources)
- Input (the interface that the packet was received on - not supported)
- Output (the index number of the interface that the packet was sent from) (Note that your collector should have the ability via SNMP to resolve index numbers to physical port numbers)
- Packet Type
- Header Protocol - Ethernet ISO 88023(1)
- Packet Size (Frame Length including the FCS)
- Header Length - The sampled portion of the frame as set by the “sflow max-header-size” on page 76.11. May be shorter for small frames.
- Header Bytes

- Extended Elements Number
- Extended Elements

Note that in practice the Ethernet header is usually followed by components for the IP, TCP, and user data.

sFlow Flow Sample Fields

- Counter Sample
- Sample Type (Counter Sample, 0x0002)
- ‘Sample Sequence Number
- Sample ID (source ID index value)
- Sample Interval (as set by the “sflow polling-interval” on page 76.13)
- Counter Type (1=generic, 2=Ethernet)

Generic Interface Counters

- ifIndex
- ifType
- ifSpeed
- ifDirection (0=unknown, 1=full-duplex, 2=half-duplex, 3=in, 4=out
- ifStatus
- InOctets
- InUcastpackets
- InMulticast packets
- InBroadcast packets
- InDiscarded packets (= 0)
- InPackets containing errors
- InPackets containing unknown protocols (= 0)
- OutOctets
- OutUcast packets
- OutMulticast packets
- OutBroadcast packets
- OutDiscarded packets
- OutPackets containing errors
- ifPromiscuous Mode

Ethernet Interface Counters

- dot3Stats Alignment Errors (= 0)
- dot3Stats FCS Errors
- dot3Stats Single Collision Frames (= 0)
- dot3Stats Multiple Collision Frames
- dot3Stats SQE Test Errors
- dot3Stats Deferred Transmissions (= 0)
- dot3Stats Late Collisions
- dot3Stats Excessive Collisions
- dot3Stats Internal Mac Transmit Errors
- dot3Stats Carrier Sense Errors (= 0)
- dot3Stats Frame Too Longs
- dot3Stats Internal Mac Receive Errors
- dot3Stats Symbol Errors (= 0)

The sFlow MIB

Your switch fully supports inMon’s sFlow MIB. For more information, see “Private MIBs” on page 64.78, and the website www.sflow.org/SFLOW-MIB5.txt.

Chapter 76: sFlow Commands

Command List

This chapter provides an alphabetical reference for sFlow commands.

debug sflow

This command enables sFlow® debug message logging, for sFlow sampling and polling activity on the specified ports. If no ports are specified, sampling and/or polling debug messages are enabled for all ports.

The **no** variant of this command disables sFlow sampling and or polling debug message logging on the ports selected. If no ports are specified, sampling and/or polling debug messages are disabled on all ports.

Syntax

```
debug sflow [interface <port-list>] [sampling][polling]

no debug sflow [interface <port-list>] [sampling][polling]
```

Parameter	Description
`interface`	Interface information.
`<port-list>`	The ports for which sFlow debug is to be enabled. The ports to display information about. The port list can be: ■ a switch port (e.g. `port1.2.12`) ■ a continuous range of ports separated by a hyphen, e.g. `port1.0.1-1.0.24` ■ a comma-separated list of ports and port ranges, e.g. `port1.0.1,port1.1.1-1.2.24`.
`sampling`	Debug sFlow sampling for the specified port(s).
`polling`	Debug sFlow polling for the specified port(s).

Default The sFlow sampling and or polling debug is disabled.

Mode Privileged Exec

Examples To enable sFlow debug message logging for polling and sampling on `port1.0.1` and `port1.0.7`, use the commands:

```
awplus# debug sflow interface port1.0.1,port1.0.7 sampling
        polling
```

To enable logging and polling of sFlow debug messages for polling and sampling on all ports, use the command:

```
awplus# debug sflow sampling polling
```

Related Commands show debugging sflow
no debug all

debug sflow agent

This command enables sFlow® debug message logging that is not specific to particular ports. For example, sending an sFlow datagram to the collector.

The **no** variant of this command applies the command default.

Syntax

```
debug sflow agent

no debug sflow agent
```

Default The sFlow agent debug message logging (that is not port specific) is disabled.

Mode Privileged Exec

Example To enable logging of sFlow agent debug messages, use the following command:

```
awplus# debug sflow agent
```

Related Commands show debugging sflow
debug sflow

sflow agent (address)

This command sets the sFlow® agent IP address on the switch. This address is inserted into every sFlow datagram sent from the sFlow agent switch to the sFlow collector device. The sFlow collector can then uses this address to uniquely identify and to access the switch, such as for SNMP. We therefore recommend that you change this address as little as possible.

Although the agent address can be set to any valid IPv4 or IPv6 address; we recommended that you set the sFlow® agent IP address to be the **local address**[1] that is configured on the switch. This ensures that the sFlow collector can maintain connectivity to the switch irrespective of the addition or deletion of VLAN interfaces (each of which will have its own specific IP address). Note that sFlow is rendered inactive whenever the agent address is not set.

[1] For information on local addresses and how to set them up, see the interface (to configure) command on page 12.3.

The **no** variant of this command applies its default setting to remove a configured address.

Syntax

```
sflow agent {ip <ip-address>|ipv6 <ipv6-address>}

no sflow agent {ip|ipv6}
```

Parameter	Description
<ip-address>	The IPv4 address of the switch that is acting as the sFlow agent.
<ipv6-address>	The IPv6 address of the switch that is acting as the sFlow agent. The IPv6 address uses the format X:X::X:X.

Default The sFlow agent address is unset.

Mode Global Configuration

Examples To set the sFlow agent (IPv4) address to 192.0.2.23, use the command:

```
awplus# configure terminal
awplus(config)# sflow agent ip 192.0.2.23
```

To remove the sFlow agent (IPv4) address, use the command:

```
awplus# configure terminal
awplus(config)# no sflow agent ip
```

To set the sFlow agent (IPv6) address to 2001:0db8::1, use the command:

```
awplus# configure terminal
awplus(config)# sflow agent ipv6 2001:0db8::1
```

To remove the sFlow agent (IPv6) address, use the command:

```
          awplus# configure terminal
  awplus(config)# no sflow agent ipv6
```

Related Commands show running-config sflow
show sflow

sflow collector (address)

This command sets the sFlow® agent's collector IP address and/or UDP port. This is the destination IP address and UDP port, for sFlow datagrams sent from the sFlow agent. The IP address can be any valid IPv4 or IPv6 address. Note that sFlow is rendered inactive whenever the collector address is set to 0.0.0.0 (for IPv4) or `::` (for IPv6).

The **no** variant of this command returns the IP address and UDP port values to their defaults, which will result in sFlow being deactivated.

Syntax

```
sflow collector {[ip <ip-address>|ipv6 <ipv6-address>]|
    [port <1-65535>]}
```

```
no sflow collector {[ip|ipv6]|[port]}
```

Parameter	Description
<ip-address>	IPv4 address of the remote sFlow collector.
<ipv6-address>	IPv6 address of remote sFlow collector. The IPv6 address uses the format X:X::X:X.
`port`	Destination UDP port for sFlow datagrams sent to the collector.
<1-65535>	UDP port number (default: 6343).

Default The collector address is `0.0.0.0` (which renders sFlow inactive), and the UDP port is `6343`.

Mode Global Configuration

Examples To set the sFlow collector address to `1920.2.25` and UDP port to `9000`, use the command:

```
        awplus# configure terminal
awplus(config)# sflow collector ip 192.0.2.25 port 9000
```

To remove the sFlow collector IPv4 address and leave the UDP port unchanged, use the command:

```
        awplus# configure terminal
awplus(config)# no sflow collector ip
```

To remove the sFlow collector IPv4 address and to remove the UDP port, use the command:

```
        awplus# configure terminal
awplus(config)# no sflow collector ip port
```

To set the sFlow collector address to `2001:0db8::1` and leave the UDP port unchanged, use the command:

```
        awplus# configure terminal
awplus(config)# sflow collector ipv6 2001:0db8::1
```

To remove the sFlow collector IPv6 address and leave the UDP port unchanged, use the command:

```
        awplus# configure terminal
awplus(config)# no sflow collector ipv6
```

To remove the sFlow collector IPv6 address and to remove the UDP port, use the command:

```
        awplus# configure terminal
awplus(config)# no sflow collector ipv6 port
```

Related Commands show running-config sflow
show sflow

sflow collector max-datagram-size

This command sets the maximum size of the sFlow® datagrams sent to the collector.

The **no** variant of this command resets the maximum-datagram-size to the default.

Syntax

```
sflow collector max-datagram-size <200-1500>

no sflow collector max-datagram-size
```

Parameter	Description
<200-1500>	The maximum number of bytes that can be sent in an sFlow datagram sent from the agent to the collector.

Default 1400 bytes

Mode Global Configuration

Example To set the maximum datagram size to 1200, use the command:

```
        awplus# configure terminal
 awplus(config)# sflow collector max-datagram-size 1200
```

Related Commands show running-config sflow
show sflow

sflow enable

This command enables sFlow® globally on the switch.

The **no** variant of this command disables sFlow globally on the switch.

Note that enabling sFlow does not automatically set its operational status to active. To activate sFlow the following conditions need to be met:

- sFlow is enabled.
- The sFlow agent address is set.
- The sFlow collector address is set to a valid (non zero) IPv4 or IPv6 address.
- Polling or sampling is enabled on the ports to be sampled or polled.

Syntax

```
sflow enable
```

```
no sflow enable
```

Default sFlow is disabled globally on the switch.

Mode Global Configuration

Example To enable sFlow operation, use the command:

```
        awplus# configure terminal
awplus(config)# sflow enable
```

Related Commands

show running-config sflow
show sflow

sflow max-header-size

This command sets the maximum header size of the ethernet frames sampled on a specified port. The maximum header size is measured in bytes, referenced from the first byte of the ethernet destination address and excludes the ethernet FCS fields.

If a sampled ethernet frame is longer than the maximum header size set by this command, then the frame will be truncated to the first N bytes before being placed in the sFlow datagram, where N is the maximum header size set by this command.

The **no** variant of this command resets the max-header-size to its default.

Syntax

```
sflow max-header-size <14-200>

no sflow max-header-size
```

Parameter	Description
<14-200>	The maximum number of header bytes to be sampled.

Default The max-header-size is 128 bytes.

Mode Interface Configuration

Usage The header size is measured from the first byte of the Ethernet frame MAC Destination Address.

For an environment using standard TCP IPv4 over Ethernet frames, consider the following basic protocol structure:

Ethernet header (including the 4 byte 802.1Q header component) = 18 bytes

IPv4 header = 24 bytes

TCP header = 24 bytes

Total = 66 bytes

See “Determine the max-header-size sampled data” on page 75.9 for more information on configuring this command

A similar consideration can be made for an environment using TCP IPv6 over Ethernet:

Ethernet header (including the 4 byte 802.1Q header component) = 18 bytes

IPv6 header = 40 bytes

TCP header = 24 bytes

Total = 82 bytes

Caution ⚠ In the above network scenarios:

- For IPv4 - any data existing between 66 bytes and the value set by this command will be included in the sFlow packet samples. For example, with the default of 128 applied, up to 128-66=62 bytes of user data could be included in the sFlow datagram samples sent between the Agent and the Collector.
- For IPv6 - any data existing between 82 bytes and the value set by this command will be included in the sFlow packet samples. For example, with the default of 128 applied, up to 128-82=46 bytes of user data could be included in the sFlow datagram samples sent between the Agent and the Collector.

Note that the agent-to-collector datagrams contain their own UDP headers, which are outside this calculation.

Example To set the maximum header size to 160 bytes for ports 1.0.1 and 1.0.7, use the commands:

```
       awplus#  configure terminal
awplus(config)#  interface port1.0.1,port1.0.7
awplus(config-if)#  sflow max-header-size 160
```

Related Commands

show running-config sflow
show sflow interface
sflow max-header-size

sflow polling-interval

This command sets the sFlow® counter polling interval (in seconds) for the specified ports. A value of 0 disables polling. A counter sample is taken every N seconds where N is the value set by this command.

The **no** variant of this command applies the default.

Syntax

```
sflow polling-interval {0|<1-16777215>}

no sflow polling-interval
```

Parameter	Description
0	Disable polling (the default).
<1-16777215>	The polling interval in seconds.

Default The polling-interval is 0 (polling disabled).

Mode Interface Configuration

Example To set the polling interval to 60 seconds for ports 1.0.1 and 1.0.7, use the following commands:

```
         awplus# configure terminal
 awplus(config)# interface port1.0.1,port1.0.7
awplus(config-if)# sflow polling-interval 60
```

Related Commands show running-config sflow
show sflow interface

sflow sampling-rate

This command sets the mean sFlow® sampling rate for the specified ports. Sampling occurs every N frames (on average), where N is the rate value set via this command. The sampling rate applies to ingress and egress frames independently. For example, a value of 1000 will sample one frame in every 1000 frames received. One in every 1000 frames sent from the specified port. A value of 0 disables sampling on the specified ports.

The **no** variant of this command applies the default.

Syntax

```
sflow sampling-rate {0|<256-16777215>}

no sflow sampling-rate
```

Parameter	Description
0	Sets the default.
<256-16777215>	The sampling rate N, measured in ethernet frames.

Default The sampling-rate is 0 (sampling disabled).

Mode Interface Configuration

Example To set the sampling rate to 500 for ports 1.0.1 and 1.0.7, use the commands:

```
         awplus# configure terminal
 awplus(config)# interface port1.0.1,port1.0.7
awplus(config-if)# sflow sampling-rate 500
```

Related Commands show running-config sflow
show sflow interface

show debugging sflow

This command displays sFlow® debug settings for agent operation, and for sampling and polling on specific interface ports. If no interface ports are specified, sampling and polling will be applied to all ports.

Syntax `show debugging sflow [interface <port-list>]`

Parameter	Description
`interface`	The interface information.
`<port-list>`	The ports for which the sFlow debug settings are to be shown. The ports to display information about. The port list can be: ■ a switch port (e.g. `port1.2.12`) ■ a continuous range of ports separated by a hyphen, e.g. `port1.0.1-1.0.24` ■ a comma-separated list of ports and port ranges, e.g. `port1.0.1,port1.1.1-1.2.24`.

Mode User Exec and Privileged Exec

Example To display sFlow debug settings on the agent, and for sampling and polling on ports 1.0.1 to 1.0.9, use the command:

```
awplus# show debugging sflow interface port1.0.1-1.0.9
```

Output

Figure 76-1: Sample obtained for an sFlow agent

```
awplus# show debugging sflow interface port1.0.1-1.0.9

sFlow Agent Debug:    Enabled

            Sampling      Polling
Port         Debug         Debug
-------------------------------
1.0.1       Enabled       Enabled
1.0.2       Enabled          -
1.0.3          -             -
1.0.4          -             -
1.0.5          -             -
1.0.6          -          Enabled
1.0.7          -             -
1.0.8          -          Enabled
1.0.9          -          Enabled
```

To display sFlow debug settings for all ports, use the command:

```
awplus# show debugging sflow
```

Related Commands show running-config sflow
show sflow interface

show running-config sflow

This command displays the running system information specific to the sFlow feature.

Syntax

```
show running-config sflow
```

Mode

Privileged Exec and Global Configuration

Example

To display the sFlow running configuration information, use the command:

```
awplus# show running-config sflow
```

Output

Figure 76-2: Example output from the **show running-config sflow** command

```
awplus#sh run sflow
!
sflow agent ip 192.0.2.33
sflow collector ip 192.0.2.65
sflow collector max-datagram-size 1200
sflow enable
!
interface port1.0.11-port1.0.22
 sflow sampling-rate 512
```

Related Commands

show running-config

show sflow

This command displays non-port-specific sFlow agent configuration and operational status.

Syntax `show sflow`

Mode Privileged Exec

Example To display sFlow configuration and operational status, use the command:

```
awplus#  show sflow
```

Output

Figure 76-3: Example output from the **show sflow** command

```
sFlow Agent Configuration:                    Default Values
  sFlow Admin Status ........ Disabled        [Disabled]
  sFlow Agent Address ....... [not set]       [not set]
  Collector Address ......... 0.0.0.0         [0.0.0.0]
  Collector UDP Port ........ 6343            [6343]
  Tx Max Datagram Size ...... 1200            [1400]

sFlow Agent Status:
  Polling/sampling/Tx ....... Inactive because:
                                - sFlow is disabled
                                - Agent Addr is not set
                                - Collector Addr is 0.0.0.0
                                - Polling & sampling disabled
                                  on all ports
```

Table 76-1: Parameters in the output of the **show sflow** command

Output Parameter	Description
`sFlow Admin Status`	Whether sFlow agent operation is administratively enabled.
`sFlow Agent Address`	The sFlow agent IPv4 or IPv6 address for the device. sFlow is rendered inactive whenever the agent address is not set.
`Collector Address`	The IPv4 or IPv6 collector address to which sFlow datagrams are sent. sFlow is rendered inactive whenever the collector address is set to 0.0.0.0 or 0:0::0.0.
`Collector UDP Port`	The UDP port on the collector to which sFlow datagrams are sent.
`Tx Max Datagram Size`	The maximum size of the sFlow datagrams sent to the collector.
`Polling/sampling/Tx`	Whether sFlow sampling and/or polling (and hence sFlow datagram transmission) are active. If inactive the reasons are listed.

Related Commands show running-config sflow
show sflow interface

show sflow interface

This command displays sFlow agent sampling and polling configuration for specified ports.

Syntax `show sflow interface <ifrange>`

Parameter	Description
<ifrange>	The interface range.

Mode Privileged Exec

undebug sflow

This command applies the functionality of the no debug sflow command.

Part 8: Virtual Chassis Stacking

Chapter 77: Stacking Introduction

VCStack Introduction

This chapter describes Virtual Chassis Stacking (VCStack), its features, and basic connection examples. For detailed descriptions of the commands used to configure VCStack, see Chapter 78, Stacking Commands.

A Virtual Chassis Stack (VCStack) is a group of physically separate switches that are connected so as to function as a single logical switch. In order to function as a VCStack, its component switches are connected using high-speed stacking links.

Features of Virtual Chassis Stacking

Creating a VCStack greatly eases network management, because you can configure all the stacked devices via a single IP address. Creating a VCStack will often eliminate your need to configure protocols such as VRRP and Spanning Tree. VCStack also enables you to create highly resilient networks. This resiliency can be applied in several ways.

Within the stack itself, switch interconnection is via two links. The second link is able to provide an alternative data path, thus the stack will continue to function if a single switch fails. Degraded performance might occur however, due to the reduced VCStack bandwidth.

User ports can also be made extremely resilient by utilizing link aggregation. Aggregated links can span ports, modules, and even switches within the stack. Creating aggregated links that span multiple switches within a stack creates an extremely resilient configuration. Communication will still exist even if a switch and its aggregated ports fail. Refer to Figure 77-7 on page 77.7.

Note IPv6 is only supported in the stand-alone mode. It is not supported in VCStack configurations.

VCStack Capable Switches

VCStack is supported on the following switch types:

- x900-24XT, x900-24XS, x900-24XT-N, x900-12XT/s
- SwitchBlade® x908
- x610 series
- x510

Note You can only create VCStacks using switches from within the same product group; i.e. all x510 switches, or all x900 series switches.

The Physical Stack

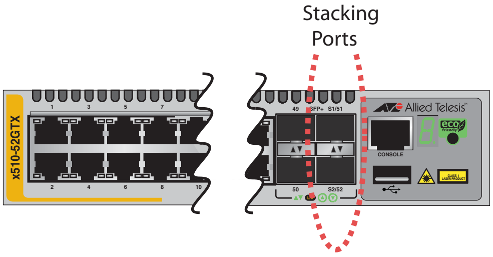
A VCStack comprises between two and four individual stack members interconnected via high-speed stacking links. As the stack forms, its switch members elect one of them to become the primary stack member called the **stack master**, (displayed in the show commands as the **active master**). The remaining switches then become ordinary members of the stack, and are referred to as backup members.

A stack can comprise from two to four individual stack members interconnected via high speed stacking links connected to SFP ports 27 and 28. A stack always has a primary stack member called the **stack master**

VCStack, cables and connections

Stack members are interconnected via the SFP+ ports shown in Figure 77-1. Note that stacking cables should connect from port SFP/51 on one stack member, to port SFP/52 on the other (or fom port SFP/27 on one stack member, to port SFP/28 on the other).

Figure 77-1: x510 Stacking Ports



x510 Stacking-Ports.eps

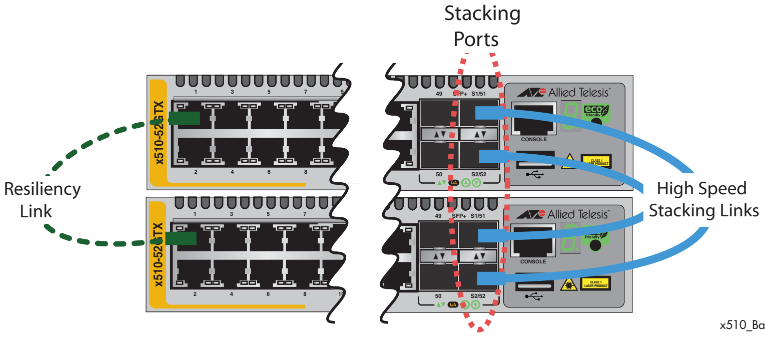
Long Distance Stacking

You can extend the distance between stacked units up to the maximum supported by the particular SFP+ you are using. This capability enables you to create a stack of up to 4 geographically separated x510 switches as a single stack.

Two Switch Stack Configuration

This configuration, shown in Figure 77-2 switches, uses two switches that are connected back to back via two high-speed stacking links. Note that stacking ports labeled S1/51must connect to stacking ports labeled S2/52. Although in this configuration the stack can still function using only a single high-speed stacking link, we recommend using both stacking links as shown.

Figure 77-2: x510 Back-to-Back Topology



x510_Back-to-BackStack.eps

Resiliency link

The function of the resiliency link is to provide additional stack status information to enable the stack members to more accurately decide whether it is appropriate for one of them to take over the role of stack master if the existing master fails. This link carries no network data. See "Stack Resiliency Link" on page 77.13.

A resiliency link operates using a designated VLAN running over switch port connections between each stack member. This connectivity method is covered in the next section of this chapter.

Resiliency link configurations via switch ports

Two resiliency-link configurations that use switch ports are shown below. Figure 77-3 on page 77.4 shows the resiliency link connecting in a ring topology, whilst Figure 77-4 on page 77.5 shows the resiliency link connecting to the switch ports via a network hub. In both configurations, the resiliency link connections are made using a designated VLAN running over switch-port connections between each stack member. For more information on using the resiliency link commands see: stack resiliencylink command on page 78.34 and switchport resiliencylink command on page 78.40.

Figure 77-3: Resiliency link connecting to switch ports over the ResiliencyLink VLAN

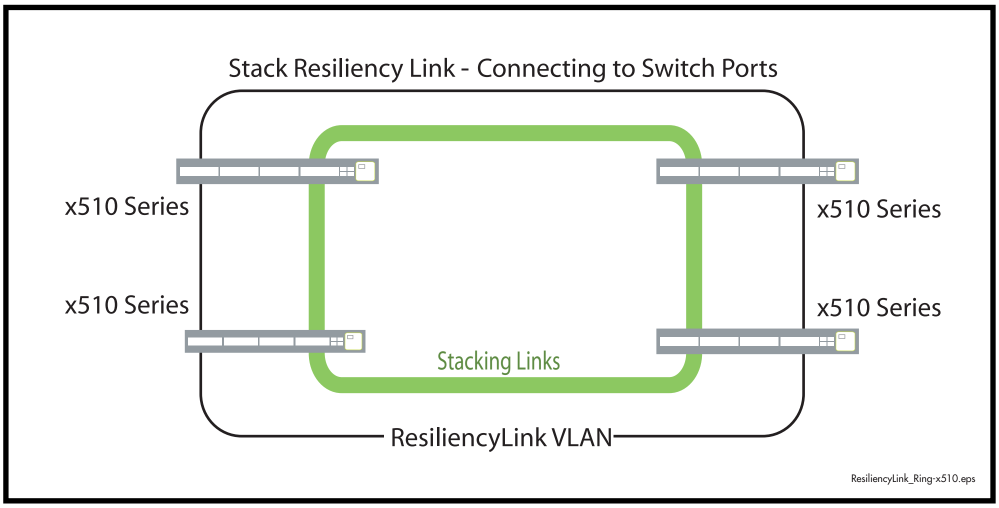

Stack Resiliency Link - Connecting to Switch Ports
x510 Series
x510 Series
x510 Series
x510 Series
Stacking Links
ResiliencyLink VLAN
ResiliencyLink_Ring-x510.eps

Figure 77-4: Resiliency link connecting to switch ports over the ResiliencyLink VLAN using a network hub.

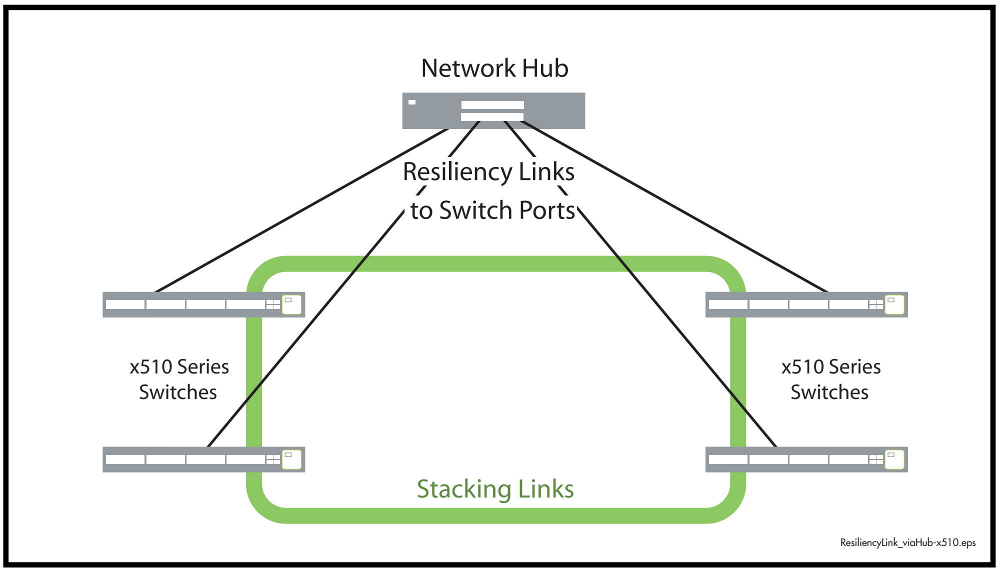
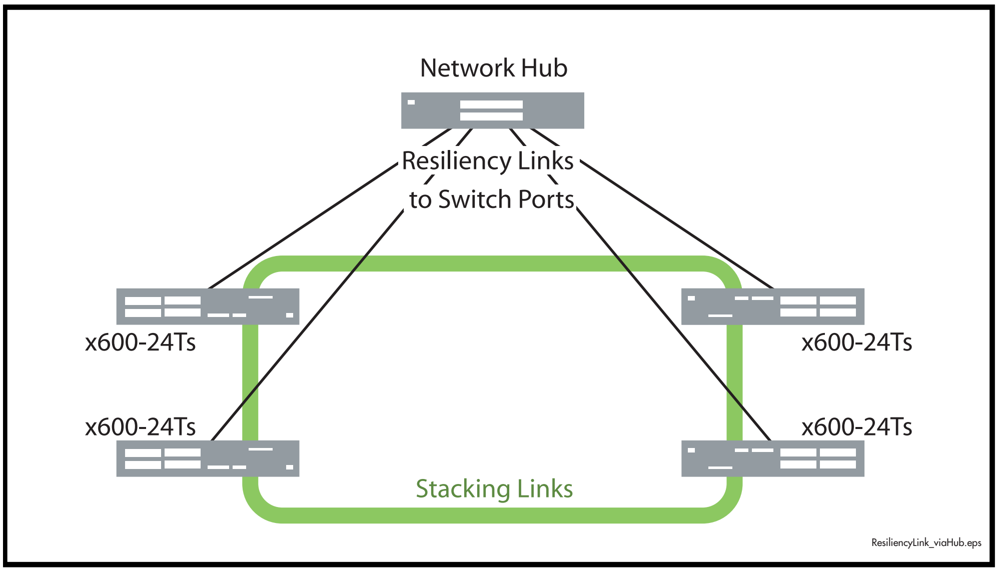


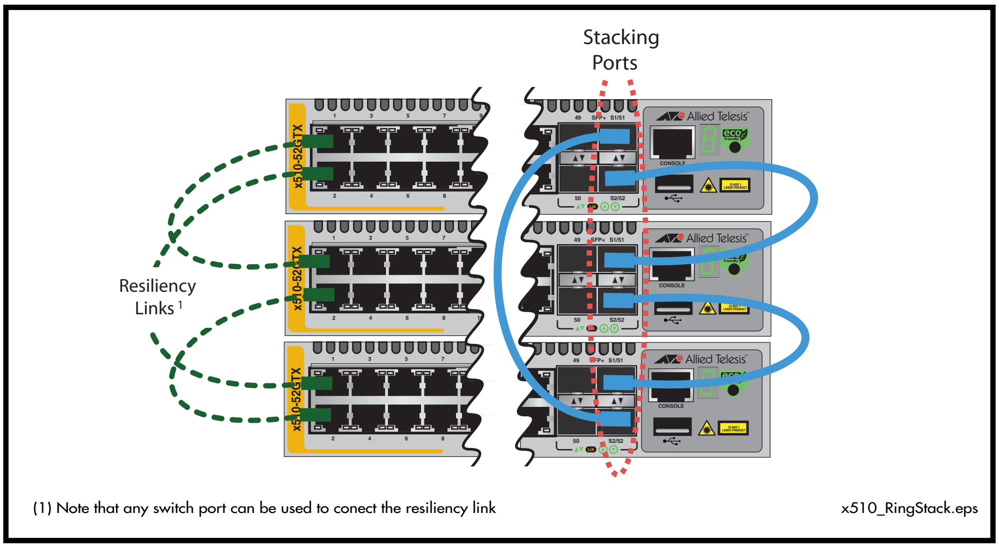
Figure 77-5: Resiliency link connecting to switch ports over the ResiliencyLink VLAN using a network hub.



Ring configuration

A VCStack using x510 switches can comprise up to four stack members connected in a ring topology. Figure 77-6 shows a ring comprising three stacked x510 series switches. Because alternate paths are provided between the stack members' stacking links, this topology offers a very resilient configuration.

Figure 77-6: VCStack Ring Topology Using x510 Switches



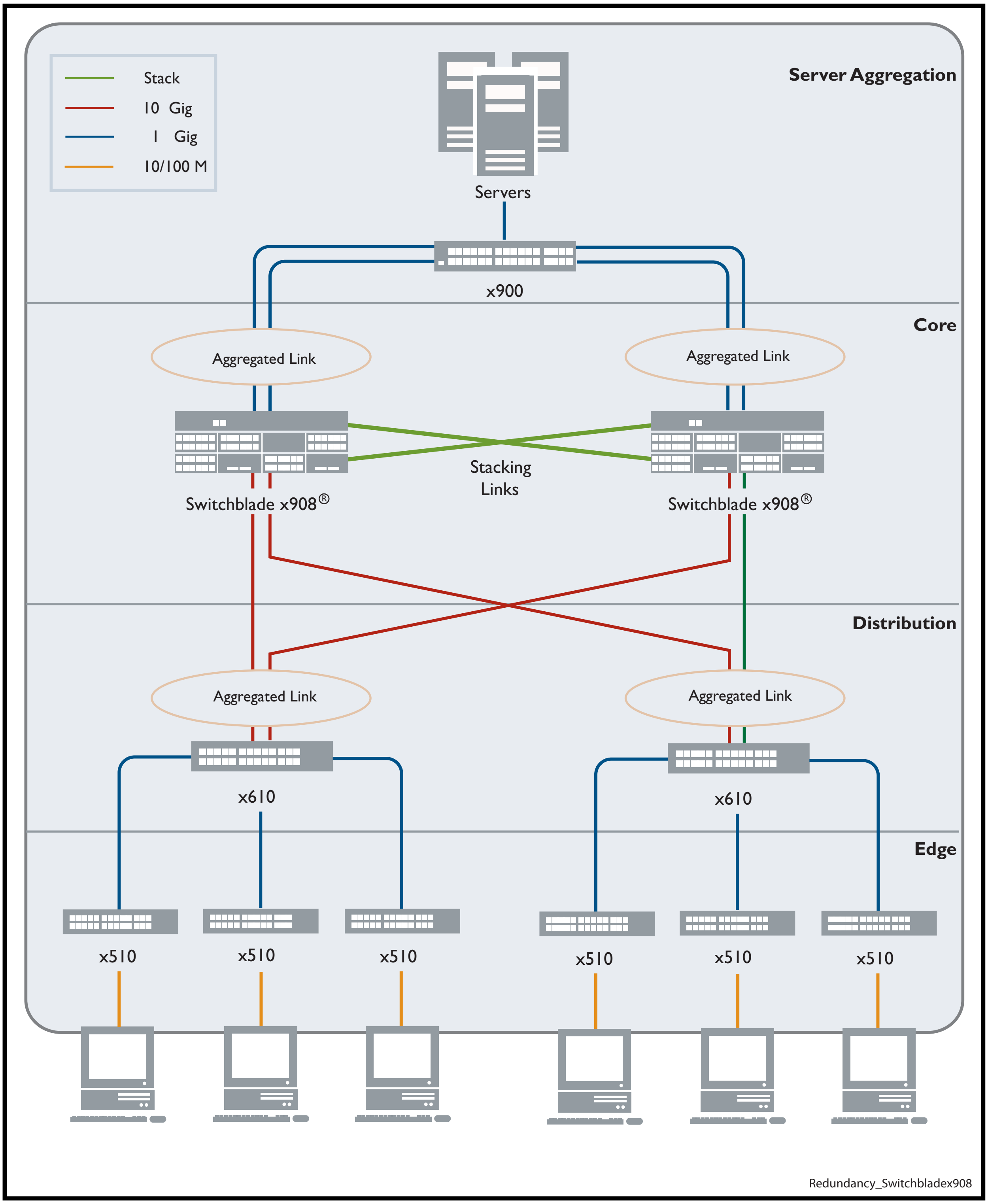
(1) Note that any switch port can be used to conect the resiliency link

x510_RingStack.eps

Resilient Stacked Topology

Where network connectivity uptime is a major criteria, you can use virtual chassis stacking to create highly reliable network configurations. The network shown in Figure 77-7 employs duplicate links and switches to create a stacked network that offers extremely reliable user connectivity.

Figure 77-7: VCStack Resilient Stacked Topology Example



This network employs two SwitchBlade® x908 switches to form an expandable network core. These switches are stacked and so appear as a single logical switch (note that smaller switches such as the x610 can be also be used to form the stacked core).

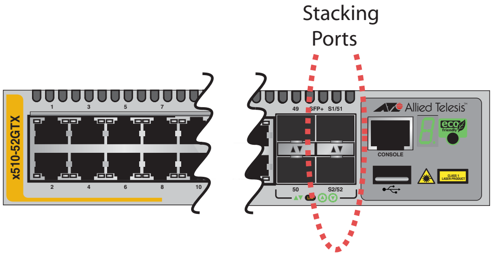
This network topology supplies multiple dual connections to a number of downstream distribution x610 switches that can in turn connect to user devices via a number of x510 switches. Similarly, the dual network paths provide very reliable connectivity to the server aggregation portion of the network.

Employing link aggregation rather than spanning tree to manage the parallel paths, enables the bandwidth of both data links to be utilized under normal conditions, whilst enabling a single data link to operate should its partner link fail.

Stack Formation

As previously mentioned, a VCStack always contains a stack (active) master plus a number of stack (backup) members. To be part of a stack, a switch must connect to other potential stack members via dedicated stacking ports. See "x510 Stacking Ports".

Figure 77-8: x510 Stacking Ports



x510 Stacking-Ports.eps

Once the switches have been physically connected to form a stack, powering all the members on automatically sets off a number of processes that enable the stack members communicate with other potential stack members and form themselves into a VCStack.

The Role of the Stack (Active) Master

In addition to being a member of its VCStack, the stack (active) master manages functions such as software version control and distribution, routing processing, and network management.

Selecting the stack (active) master

The stack members are able to automatically decide (elect) which switch will become the stack (active) master. This election process is based on two components:

1. The numeric value assigned using the stack priority command on page 78.31

2. The stack member's MAC address.

For both components, the lower the number the higher the priority. To set the stack priority, run the stack priority command. Note that changes to these settings will not take effect until the next master re-election. To display these components run the show stack command on page 78.21.

The master is the switch that has the lowest 'numeric value' set by the stack priority command, or if the 'priority settings' are equal, the master will be the switch with the lowest MAC address. When a stack member is initially booted, its priority value defaults to 128. Therefore if all switches retain their defaults, the stack master will be determined sole by MAC address comparison.

The stack also assigns a Stack Member-ID number to each member. This number provides a unique reference number for switches within the stack, but this number plays no part in selecting the stack master. The Stack Member-ID is the entity used as the first digit of the three component port identifier numbers. For example, port number 2.0.14 has the Stack Member-ID of 2.

Note This last point is an important one to remember when using configuration scripts. You should ensure that you modify your configuration scripts to match any changes you have made to the Stack Member-ID assignments.

Note The ability to independently set both a stack member's priority and its ID means that the stack master does not need to have an ID of 1; although configuration is simplified by arranging for ID 1 to be the device with the lowest priority value - and thereby forcing it to be the stack master. If you create a stack using new switches, the following (simplified) process should ensure that the master member has an ID of 1.

New switches are shipped with a Stack Member-ID of 1 and a priority of 128. If four such switches are created as a stack, the switch with the lowest MAC address will be selected to be the stack master (because all priority settings are 128). The remaining three stack member devices will then reboot. The stack master does not reboot and retains its Stack Member-ID of 1.

You can change the Stack Member-ID by using the stack renumber command on page 78.32.

Common Stack Configuration

Once the switches have configured themselves into a VCStack, they all share the same configuration information and startup scripts.

Stack Management VLAN

Managing the stack is the same as managing an individual switch. You can connect to the asynchronous console port of any stack member, or you can set an IP address on a network VLAN (for example, VLAN 1) and use SSH for remote access.

As the switches form themselves into a stack, each switch creates a common stack management VLAN and a management IP address. Both the VLAN ID and the IP address are internal entities that are used between the stacked switches, via the assigned stacking ports, and therefore do not appear on the user network.

Initially the stack assigns the default VLAN tag ID of 4094 to the management VLAN, and assigns an IP address from the subnet 192.168.255.0 / 28 to this VLAN as the management IP address. Once the stack has formed, you can change both these settings. To change the VLAN ID use the stack management vlan command on page 78.30. To change the management IP address use the stack management subnet command on page 78.29. Note however, that you must keep the 28 bit subnet mask, (/28 or 255.255.255.240). Also note that because the stack's internal address mapping tables will register the management VLAN ID and the management IP address, these must be unique across the stack's internal and external network. To view the current settings for the stack management VLAN ID and IP address, use the show stack command on page 78.21.

Stack member identification

Running commands on specific stack members

In some situations, you may want to obtain information that is specific to a particular stack member. To achieve this you can use the **remote-command** feature. For example, to see the processes running on stack member 3, you can run the following **remote-command**:

```
awplus# remote-command 3 show process
```

For more information and options on this feature, see the remote-command <1-4> show command on page 78.9 of this manual.

Running QoS within a VCStack

In general you can apply the same principles when configuring QoS on a VCStack as you would for single switch; however there are a few specific changes that you will need to make.

Switches within a VCStack exchange their stack management information and user data over their high speed inter-stacking links. The stack management information is pre-assigned to the egress queue 7. This is the highest value queue, and (in a stacked configuration) its traffic should not be shared with any user data. However, any CoS tagging of 7 applied to the incoming data will automatically be assigned to queue 7 as it crosses the internal stacking links. You will therefore need to reconfigure your CoS to Queue settings to ensure that no user data is sent to queue 7.

To prevent this from happening, we recommend that you make appropriate changes to your queue settings (mappings) to reflect the stacking requirement previously described. For more information on this topic, see "Mapping CoS tags to traffic types" on page 35.12.

This process should include (but not be limited to) running the following command to ensure that any remaining user packets still carrying a CoS 7 tag, will be mapped to egress queue 6.

To remap priority CoS traffic to egress queue 6, run the following commands:

```
        awplus# config terminal
awplus(config)# mls qos map cos-queue 7 to 6
```

Stack Member Failure and Recovery

Fixed or Virtual MAC Addressing

A VCStack operates using a single **virtual** MAC address. This address is configurable by using the stack virtual-mac command on page 78.38.

Enabling the stack virtual-mac

When the stack virtual-mac command is enabled, the stack uses a virtual MAC address that is either manually entered, or has been randomly selected from an allocated pool of MAC addresses. The stack will then always use this MAC address even if the stack master fails or is removed from the stack. In this situation, the new elected master will still retain the originally configured virtual MAC address.

The virtual MAC address will be used for all external ports, and VLAN interfaces, except the management VLAN. Although each individual switch in the stack retains its own native MAC address; this is only used over the stack management VLAN.

Note If one stack member has the virtual MAC address feature enabled and another has the virtual MAC address feature disabled then they will be able to form together as a stack. From master election onwards, the stack master's virtual MAC address setting will be used by the rest of the stack.

Virtual MAC format and value

The virtual MAC address is selected from within the range 0000.cd37.0000 to 0000.cd37.0FFF. This can be considered as a MAC prefix component of 0000.cd37.0xxx.
Where xxx is called the stack virtual-chassis-ID, and has the range 000 to FFF.

Manually selected virtual address

To manually select a virtual MAC address you enable the stack virtual MAC feature by using the commands:

```
        awplus# configure terminal
awplus(config)# stack virtual-mac
```

Then configure the stack virtual-chassis-id command on page 78.37 to set a stack virtual-chassis-ID of your chosen value - entered as a decimal number within the range 0 to 4095. The value 120 is used in the following example:

```
        awplus# configure terminal
awplus(config)# stack virtual-chassis-id 120
```

Automatically selected virtual address

If you set the stack virtual-mac command without entering a value for the stack virtual-chassis-ID, the switch will randomly select a virtual-chassis-ID from the allocated range.

Disabling the stack virtual-mac

When the stack virtual-mac command is disabled, the stack will use the MAC address of the current Master. If the stack master fails, the stack MAC address changes to reflect the new master's MAC address. If the stack MAC address does change, ARP tables of devices on the network will update to reflect the change in MAC address via ip gratuitous-arp-link command on page 25.26.

Stack Resiliency Link

The purpose of the resiliency link is to provide the stack members with status information that allows them to detect whether the stack master is still operational after a stack failure occurs.

Using the resiliency link, a stack member can differentiate between the master suffering a power-down or a software lock-up, where the master is offline, as opposed to a stacking-link failure, where the master is still online but connectivity over the stacking cables has been lost.

This enables the other stack members to either operate in the fall-back Disabled Master mode, or to re-elect a new stack master. The "State Change Table" on page 77.14 shows how the stack members respond to various problems occurring on the master node.

Stack recovery states

The following state-change-table shows separately the stack member failure conditions and recovery actions in situations when the resiliency link is present and when it is absent.

Note This table is only valid for a two unit stack. For explanations of state behavior in larger stacks see the following sections:

- "Disabled Master" on page 77.16
- "Stack DMM Operation" on page 77.17
- "Stack Resiliency Link" on page 77.13

Table 77-1: State Change Table

Event on Master Node	**Reaction on Master**	**Reaction on Stack Member (With Resiliency Link)**	**Reaction on Stack Member (Without Resiliency Link)**
Both stack links removed	No change	Moves to Disabled Master state.	Elect new master[2]
Hardware reset (or fault)	Reset / offline	Moves to Disabled Master state.	Elect new master[2]
Run the **no stack enable** command[3]	No change	Moves to Disabled Master state.	Not allowed Displays Error Message
Software application problem (lock-up or continual crashes)	Reboot as stack member	Re-elect master	Re-elect master
Software crash or lock-up	Frozen[3]	Elect new master	Elect new master
Power-down or PSU failure	Powered down	Re-elect master	Re-elect master
Event on Stack Member Node	**Reaction on Master**	**Reaction on Stack Member**	**Reaction on Stack Member**
Both stack links removed	No change	Disabled Master[1]	Re-elect master [2]
Hardware reset (or fault)	No change	Reset/offline	Reset/offline
Run the **no stack enable** command[3]	No change	Disabled Master	Disabled Master
Software application failover (lock-up or continual crashes)	No change	Re-boot as backup member	Reboot as stack member
software crash or lock-up	No change	Frozen[3]	Frozen[3]
Power-down / PSU failure	No change	Powered Down	Powered down

[1] When a backup member becomes the Disabled Master it will first disable all its switch ports, then activate any triggers specified with the type stack disabled-master command that have been configured.

[2] The stack member assumes the role of stack master. In specific situations this condition could result in a stack containing two masters. This would present problems with network management and the control of links that were previously aggregated.

[3] If the backup member's ports are still up, this may cause downstream switches with trunked ports to operate incorrectly.

Stack Failure Recovery

If the stack master either fails, or is removed, the other stack members will elect a new stack master. See the Disabled Master Monitoring (DMM) and the Disabled Master sections for more information. Alternatively, you can manually configure a trigger with the type stack disabled-master command on page 72.29 to activate on a stack member if it becomes the disabled master.

Table 77-1 shows how the stack backup members would respond to various problems occurring on the stack master stack.

Note When VCStack is used with EPSR, the EPSR **failovertime** must be set to at least 5 seconds to avoid any broadcast storms during failover. Broadcast storms may occur if the switch cannot failover quickly enough before the EPSR **failovertime** expires. See the epsr command for further information about the EPSR **failovertime**. See the reboot rolling command for further information about stack failover.

Stack Separation and Recovery

Stack stubs occur when a fault results in the stack splitting into two, with one of the stack members taking on the role of stack master. Where the stack master is still active after a fault, and other stack members are not aware that the stack master is still active, the result can be two independently operating stacks, or stubs.

When two stub stacks are reconnected, a dual master situation will be detected, and the console log will display the message that a 'duplicate master' was detected. This situation results in the re-election of the stack master based upon the lowest Priority ID, or, where both members have the same Priority ID, the lowest MAC address. The 'losing' master and other prospective stack members will then reboot and join the new stack as backup stack members.

Note Stubs are unlikely to cause network connectivity problems if a resiliency link is used.

Stack Maintenance

Adding a stack member

An unstacked switch can be added to an existing stack (hot-swapped in) with minimal impact on traffic. To do this, power down the new member switch, then connect its stacking ports and power on the switch. The switch will boot as a member of the stack.

Note The existing Stack Member-ID and the device MAC address will have no effect on the status of the new member switch. The stack will admit the new device as an ordinary stack member and allocate it a new Stack Member-ID if its ID is one that already exists.

However, for good practice we recommend pre-configuring the new member with settings that are appropriate for when the new switch becomes a stack member. This is to avoid unexpected situations occurring when the stack is rebooted. For example, if the new member had a priority setting that was lower than 128 and all the existing stack members were configured with the default; then, when the stack is rebooted, the new member would be elected as the stack master.

Replacing a stack member

A stack member can be replaced stack (hot-swapped) with minimal impact on stack traffic. To do this, power-down the stack member, disconnect its stacking ports. Insert the new stack member, reconnect the stacking ports and power-up the new stack member.

You can seamlessly swap a stack member switch into the stack to replace another with the same configuration. This provides a simple way to replace an out-of-service switch with minimal impact, and minimal administration requirement. You should configure the replacement switch with the same member ID as its replacement prior to inserting it into the stack.

Combining separate stacks

Small individual stacks can be combined into a single larger stack (comprising up to 4 stack members) simply by physically reconnecting the stack members and rebooting. However, the likelihood of a successful stack recombination is greatly increased if you set the stack IDs of each stack member to be unique within the combined stack that you are creating.

For example, if two individual, two member stacks are to be combined into one, four member stack, and the members of each stack had the stack member IDs of 1 and 2. Then, before you combine the stacks, you should renumber the member IDs of stack two to be 3 and 4.

Disabled Master

A properly functioning VCStack contains an (active) master and a number of (backup) member switches. The Disabled Master is a "standby" state that one of the (backup) stack members is elected to when its portion of the stack looses connectivity with the stack master. Once elected to this state the disabled master will disable all of its own ports and those of all other stack members within its stack stub. To do this, the disabled master applies the shutdown command to every switchport within its stack stub.

By disabling all its stub's switchports, the disabled master avoids potential network connectivity problems that could result from by having two stack masters using the same configuration, or two switches in separated stubs trying to share the same "logical" communications paths such as a non functioning aggregated links. The active master's ports are unaffected by this, and they will continue to forward traffic normally.

Note that status information for members of the stack stub can be accessed by logging into the disabled master, in the same way as obtaining status information for a normal stack.

A disabled master trigger also enables you to specify a script to reconfigure the disabled master on the fly, should a catastrophic failure separate the stack. This is useful to configure an alternate IP address so you can still login to the disabled master via an SSH or a Telnet connection. The trigger script should use the **no shutdown** command to re-enable any switchports needed for an SSH or a Telnet management connection. For trigger command information see the **type stack disabled-master** command on page 72.29.

Disabled Master Monitoring (DMM)

The stack resiliency link feature combined with the disabled master state offers an effective method of automatically recovering from certain types of catastrophic network connectivity situations. When a node or interlink failure separates the stack into two segments, the network enters a fragile state that is dependant on the status of the stack master. If the stack master initially remains active, but then subsequently fails during while the stack is in this separated state, then (depending on where communication break occurred) a substantial portion of the stack's connectivity could be lost. This is because their is no monitoring process to detect and restore the disabled master's switchports.

The Disabled Master Monitoring (DMM) feature rectifies this situation by "continuing" to monitor the status of the original stack master via the stack resiliency link. When the DMM feature is enabled, the disabled master can detect a failure of the original stack master (subsequent to the initial fault) within a few seconds and respond accordingly. This process is explained in the next section.

Stack DMM Operation

The operation of the DMM comprises the following steps. This process assumes that a Stack Resiliency Link is configured and enabled, and DMM is turned ON:

If a fault condition occurs, within the stack or its high speed links, that prevents stacking communication between some of the stack's members but the stack master itself remains active, then the following procedure will apply:

- The VCStack breaks into two stack segments.
- The stack segment with the active master continues to function as a smaller stack.
- The stack segment without the active master then elects a possible alternative to the stack master. Because the resiliency link indicates that the master is still active, the newly elected alternative master places itself in the **disabled master state**. In this state the switch will disable all of its own switch ports, plus those of any other members of the fragmented stack segment.
- The disabled master continues to monitor the status of the active master via the resiliency link.
- If, in addition to the original fault, the active master subsequently fails, then the resulting action will depend on whether or not there is another stack member in the segment that contains the original stack master:
  - If there is one or more stack members in the segment that contains the original stack master, then one of these members will be elected as the new stack master. The disabled master in the other segment will retain its disabled state and all ports in its segment will remain disabled.
  - If there is no other stack member in the segment that contains the original stack master, then the disabled master becomes the active master and all ports in its segment will transition from disabled, to active.
- The stack will continue in this mode until the stack fragmentation fault is rectified.

For more information about the disabled master state, see the section "Disabled Master" on page 77.16. Note that a disabled master has the same configuration as an active master, but a disabled master has all its links disabled.

To enable the DMM feature, use the commands:

```
        awplus# configure terminal
awplus(config)# stack disabled-master-monitoring
```

To disable the DMM feature, use the commands:

```
        awplus# configure terminal
awplus(config)# no stack disabled-master-monitoring
```

To show the status of DMM on the VCStack, use the command:

```
awplus# show stack detail
```

To apply a trigger upon transition from active master state to disabled master state, use the command:

```
awplus# type stack disabled-master
```

To apply a trigger upon transition from disabled master state to active master state, use the command:

```
awplus# type stack master-fail
```

Note A disabled master trigger allows you to specify a script to reconfigure the disabled master on the fly, should a catastrophic failure separate the stack. This is useful to configure an alternate IP address so you can still log in to the disabled master via an SSH or a Telnet connection. The trigger script should use the **no shutdown** command to re-enable any switchports needed for an SSH or a Telnet management connection.

Provisioning (Stack Members)

Stack member provisioning is the pre-configuration of a stack member's position ready for insertion at a later time. Provisioning enables a network administrator to pre-configure vacant stack member capacity within a VCStack, ready to be hot-swapped in at a later time. Later, when the stack member switch is physically added, its configuration is automatically applied with the minimum network disruption. Provisioning is ON by default, and cannot be disabled.

Provisioned capacity can be applied by either of the following actions:

- applying the switch provision command on page 78.39
- installing, then removing a provisionable device from its physical location, that is, a switch from its stack.

When a vacant switch is provisioned, its ports are assigned the shutdown state and are therefore not able to be activated. Applying the **shutdown** or **no shutdown** commands to a provisioned port will change only its administrative state.

Provisioned Board Classes

Provisioning introduces the concept of defined board classes. Table 77-2 on page 77.19 lists the stack member classes that have been defined for provisioning. Each board class is assigned a **class** and an appropriate **port count.** Presently no further definitions have been made for additional features such as media type, or PoE capability. This structure simplifies configuration.

Table 77-2: Provisioned Stack Member Classes

Board Classes	
Class	**Port Count**
x510-28	24 switch ports, plus 4 SFP+ ports, two of which can be used for VCStacking.
x510-52	48 switch ports, plus 4 SFP+ ports, two of which can be used for VCStacking.

Applying Hardware Provisioning

As previously mentioned, provisioning is the pre-configuration of vacant (i.e. unused) device capacity ready for device insertion at a later time.

With provisioning, you can configure stack members and their ports ready for future addition, even though currently they are not physically present:

```
awplus(config)# switch 4 provision x510-28
```

Now that the switch is provisioned within the stack—although not yet physically present—you can move on to provisionally configure the switch ports themselves. The following example sets the port speed of port 4.0.1 to be 1000 Mbps.

```
   awplus(config)# interface port4.0.1
awplus(config-if)# speed 1000
```

You can apply provisional configuration to all interface related commands. However, you cannot apply provisioning where it changes the network's physical topology. For example, you can't provision a switch as stack member 3 and then later change it - while its position is still vacant - to stack member 4. In this situation, you would need to unprovision the switch, then provision it again as stack member 4.

The following example creates a provisioned configuration that shows the association of ports with a VLAN:

```
           awplus# configure terminal
   awplus(config)# vlan database
awplus(config-vlan)# vlan 12 state enable
awplus(config-vlan)# exit
   awplus(config)# interface port2.0.1
 awplus(config-if)# switchport mode access
 awplus(config-if)# switchport access vlan 12
 awplus(config-if)# exit
   awplus(config)# interface port2.0.2-port2.0.26
 awplus(config-if)# switchport mode access
```

Provisioning Error Messages

The following error messages may appear when configuring provisioning.

Table 77-3: Provisioning Error Messages (switch x [bay y] provision)

Error Message	Comment
`Switch %d must be provisioned before bays can be provisioned.`	`%d` = stack member id
`% Switch %d (%s) has 0 expansion bays.`	`%d` = stack member id `%s` = board class
`% switch %d is already populated with %s.`	Indicates an attempt to provision a switch that is already present.
`% switch %d is already provisioned for %s.`	Indicates an attempt to provision a switch that is already provisioned.

Table 77-4: Provisioning Error Messages (no switch x [bay y] provision)

Error Message	Comment
`% switch %d bay %d is currently populated by %s`	You must remove hardware before unprovisioning

Removing Hardware Provisioning

Hardware capacity that has been previously provisioned *and is presently unoccupied* can be unprovisioned with the **no switch provision command**. This removes the provisioned configurations for hardware that has either not yet been physically added to a switch or VCStack, or has previously existed, but has been removed.

The **no switch provision command** will also delete any switch bay commands with the same unit number and all associated interfaces, as well as all configuration for that switch or XEM module.

You cannot unprovision hardware that is currently installed. A **no switch** command will not succeed if the unit/unit.bay location is currently occupied. For example:

```
awplus(config)# no switch 2 bay 8 provision
                % switch 2 bay 8 is currently populated by xem-1
```

The following example displays the output of a show stack that includes a provisioned VCStack member 3:

```
Virtual Chassis Stacking summary information
ID  Pending ID  MAC address       Priority  Status    Role
1   -           0000.cd28.5377    128       Ready     ActiveMaster
2   -           0000.cd29.95bf    128       Ready     BackupMember
3   -           0000.0000.0000    -         -         Provisioned
```

Switch ID 3 is then unprovisioned:

```
awplus(config)# no switch 3 provision
```

Run the show stack command to confirm that switch 3 has been unprovisioned.

```
awplus# show stack
```

```
Virtual Chassis Stacking summary information
ID  Pending ID  MAC address        Priority  Status    Role
1   -           0000.cd28.5377     128       Ready     ActiveMaster
2   -           0000.cd29.95bf     128       Ready     BackupMember
```

Note Ensure that you save your running configuration to your startup configuration after making any provisioning changes using **copy running-config startup-config**

Displaying Provisioned Configurations

In this respect the major difference associated with provisioning, is that interface configurations will still exist in the config files and will appear in show commands, even though a device itself may not be physically installed. This (provisioning) could result from device capability that has been preconfigured for future installation, or could result from the removal of an installed device.

The **show running-config command** includes switch commands for existing hardware, plus all non-existent, but provisioned, hardware. The following example output of the **show running-config command** illustrates how provisioned and existing hardware is displayed.

Figure 77-9: Sample display of existing and provisioned show output

```
sh running-config
.
.
switch 1 provision x908
switch 1 bay 1 provision xem-12
switch 1 bay 2 provision xem-12
switch 1 bay 3 provision xem-12
switch 1 bay 5 provision xem-12
switch 1 bay 7 provision xem-1
switch 1 bay 8 provision xem-1
switch 2 provision x908
switch 2 bay 1 provision xem-12
switch 2 bay 2 provision xem-12
switch 2 bay 3 provision xem-1
switch 2 bay 8 provision xem-1
!
interface port1.1.1-1.1.12
 switchport
 switchport mode access
!
interface port1.2.1-1.2.12
 switchport
 switchport mode access
!
interface port1.3.1-1.3.12
 switchport
 switchport mode access
.
.
.end
```

Displaying provisioned hardware status

The status, present or provisioned, appears in monitoring commands such as the **show interface brief** command. The following sample output from the **show interface brief** command shows the provisioning status of two configured stack members.

Figure 77-10: Sample **show interface brief** output showing hardware provisioning status

```
awplus#show interface brief
Interface             Status          Protocol
port2.0.24            admin up        down
port3.0.1             admin up        provisioned
port3.0.2             admin up        provisioned
```

A more detailed inspection of the provisioned port 2.8.1 is shown below. Note that the MAC address of 0000.0000.0000, which is the value applied as a placeholder for all provisioned ports. Also note that although the port is in the link DOWN state its administrative state of UP PROVISIONED means that it can be further configured. For example, it can be associated with a VLAN, or added to a link aggregation group etc.

Figure 77-11: Sample display showing provisioning status of a specific port

```
Interface port2.8.1
  Scope: both
  Link is DOWN, administrative state is UP PROVISIONED
  Thrash-limiting
    Status Unknown, Action learn-disable, Timeout 1(s)
  Hardware is Ethernet, address is 0000.0000.0000
  index 6801 metric 1 mtu 1500 mru 1522
  <BROADCAST,MULTICAST>
  VRF Binding: Not bound
  SNMP link-status traps: Disabled
    input packets 0, bytes 0, dropped 0, multicast packets 0
    output packets 0, bytes 0, multicast packets 0 broadcast
pks0
```

Provisioning and Configuration Management

A benefit of provisioning is configuration settings are no longer dependant on the existence of hardware devices. When a device is removed, all the interfaces for that device are shutdown and its provisioning status is set. This means that you can add or remove physical hardware without affecting your network interfaces. Of course when ports go down (i.e. are physically removed) there will be other changes to network configuration, as protocols may re-converge or, for example, routes may be removed etc.

Switches within a VCStack (or XEMs within a switch) can be hot-swapped without the need for reconfiguration.

The configuration of a newly inserted device that matches the provisioned board-class is achieved on a best-effort basis. For example inserting a non-POE switch into a stack member location configured for PoE will result in the failure of the PoE configuration commands.

Take care that your provisioned configurations, match with the type of hardware that you plan to install. For example, the XEM-12 configuration shown below has the port speed set for one of its ports:

```
   awplus(config)# switch 2 bay 4 provision xem-12
   awplus(config)# int port2.4.1
awplus(config-if)# speed 10
```

This will be fine if a XEM-12T is installed; however installing a XEM-12S (a device that cannot run its ports at 10 Mbps) would result in an error condition.

Software Version Auto Synchronization

Introduction

Different software releases have functional and operational differences between them. To maintain consistent behavior across the stack, all new member switches must be running the same software release before they can fully join the stack.

Manually upgrading the software release of each new stack member that joins a stack would be a cumbersome process. The VCStack software version auto synchronization feature automates this process by ensuring the same software release is used on all stack members, and automatically upgrading stack members where required.

How auto synchronization works

Software version comparison

When the stack is formed, it elects one of its switches to become the master. The software release running on the stack master will then become the software version used throughout the stack. After a master is elected, all the stack members compare their current software version with the version that is running on the stack master.

If the comparison process detects differences between software versions, the software version synchronization feature will automatically copy the master's software release onto the appropriate stack members. Once the software release has successfully been copied, this version will become the boot software for that particular stack member, which will then reboot in order to load the new software release.

If a software version running on a stack member is incompatible with that running on the master, and software-auto-synchronization is turned off, then that switch will be removed as a stack member. See stack software-auto-synchronize command on page 78.36.

From release 5.4.1 onwards, when auto-synchronization upgrades a stack member, the member's current running software will be set as the backup software release. If there are any problems loading the new software, then the backup software release will be used to recover.

If the stack member does not have enough free Flash memory space for the new release, then the new release will replace up to two older release files in Flash memory, which is determined by software build dates of the older release files. The oldest release files are replaced first.

Auto synchronization limitations

Because the stack master's software version gets applied to the rest of the stack, care must be taken to ensure the correct switch is elected master. If the master is running an older software release, then software version auto synchronization may actually downgrade the software releases running on other stack members. For configuring which stack member becomes the master, see the stack priority command on page 78.31.

Software auto-synchronization will not work if stack members are booting using either one-off boot or from TFTP or ymodem. In these situations, any stack members running different software will boot as standalone devices.

If software-auto-synchronization is configured as off for a stack member that is running a different software release to the master, by applying the command, **no stack software-auto-synchronize,** then that switch will boot as a standalone device. For more information, see the stack software-auto-synchronize command on page 78.36.

Incompatible software releases

The auto-synchronization feature will not always work if there have been significant VCStack or system changes between the two different software releases. The VCStack discovery of other stack members uses an internal 'stack S/W version' to detect compatibility between builds.

If the VCStack software between two stack members is incompatible, the software auto-synchronization feature will not work. Instead, a "incompatible stack S/W version" log message will be displayed and both stack members will boot as standalone devices. This is an undesirable situation because both devices may load the same configuration file, which could cause network conflicts. In order to avoid this situation when upgrading the stack to a new major release, ensure the 'boot system' command succeeds.

In general, the software-auto-synchronize feature will always work between maintenance releases, such as between 5.3.2-0.1 and 5.3.2-0.2, but may not work between major releases that have new VCStack features, such as between 5.2.2-0.9 and 5.3.2-0.2.

Upgrading stack software reliably

When upgrading a stack to a new software release, the boot system command on page 7.8 will automatically synchronize the new software release across all stack members. If there is insufficient file space on a backup member, the boot system command has an interactive mode that prompts you to delete old releases to free up file space. However, if you choose not to delete any release files, or if Flash space is taken up with other types of files, then the boot system command can fail to set the preferred release on the backup member. If this situation occurs, it is recommended to manually free up file space on the stack member and then reenter the boot system command.

If you are unsure which files to delete, the following process may assist you.

```
  awplus# remote-login 2
awplus-2# enable
awplus-2# dir
```

Use the remote-login command to login to the backup member with insufficient free file space, in this example member-2.

Look for any .rel (release) files, .jar (GUI), or .tgz and .gz (diagnostic) files that are no longer needed and use the delete command to remove them from the back-up member's file system.

Alternatively, you can use the file system commands directly from the master's console prompt using the filepath of the backup member's Flash. Substitute **awplus** for the hostname in the configuration, and use **awplus-3** for stack member-3, and so on.

```
dir awplus-2/flash:*
delete awplus-2/flash:/x510-5.4.2A.rel
```

Stack License Management

You can manage feature licenses across all members of a stack concurrently. This means that a feature license for a product family, which is defined as a set of devices containing variants of the same product type that are able to be stacked together, can be enabled or disabled on all members of the stack at the same time. Previously, you had to configure each member of the stack individually.

Feature licenses across a stack can be managed either via the CLI with the license member command or via SNMP using the AT-LICENSE-MIB Enterprise MIB.

The stack does not have to be restarted when a feature license is enabled or disabled. When a new feature license is enabled across the stack some protocol modules may need to be restarted before the license is activated. A warning is given about the need to restart the protocol modules and a "y/n" prompt is given to proceed with the license installation. If SNMP is used to enable a feature license, the affected protocol modules are restarted automatically.

When you remove a feature license across all members of the stack via the CLI you are prompted for confirmation before the license is disabled. If SNMP is used to disable a feature license no warning is generated.

To maintain consistent behavior across the stack, all member switches should have the same feature licenses enabled. However, you can enable or disable a feature license on a single stack member if required. Note that doing so could result in the stack failing to operate correctly or in the stack separating. If you enable or disable a feature license on a single stack member via the CLI a warning message is generated. A warning message is not generated if you manage the stack via SNMP.

A license key can be purchased that includes a limit on the number of switches that it can be applied to. If the license is applied to a stack that has more members than the license is valid for via the CLI, a warning message is generated and the event is logged.

For feature licenses, contact your authorized distributor or reseller. If a license key expires or a proper key is not installed, some software features will not be available.

Note See the AlliedWare Plus^{TM} datasheet for a list of current feature licenses available by product, and the AlliedWare Plus^{TM} How To notes for information on obtaining them.

To enable a feature license on all stack members, use the command:

```
awplus# license <name> <key> member all
```

To enable a feature license on a specific stack member, use the command:

```
awplus#  license <name> <key> member <1-8>
```

To disable a feature license on all stack members, use the command:

```
awplus# no license <name> member all
```

To disable a feature license on a single stack member, use the command:

```
awplus# no license <name> member <1-8>
```

To display detailed information about feature licenses on all stack members, use the command:

```
awplus# show license [<name>] member all
```

To display brief information about feature licenses on all stack members, use the command:

```
awplus# show license [<name>] brief member all
```

To display detailed information about feature licenses on a single stack member, use the command:

```
awplus# show license [<name>] member <1-8>
```

To display brief information about feature licenses on a single stack member, use the command:

```
awplus# show license [<name>] brief member <1-8>
```

Chapter 78: Stacking Commands

Introduction

This chapter provides an alphabetized reference for each of the Stacking commands. Also note the following stacking trigger commands that are documented in the Triggers chapter:

type stack disabled-master command on page 72.29
type stack master-fail command on page 72.30
type stack member command on page 72.31
type stack link command on page 72.32

In addition to the stacking commands shown in this chapter, stacking content also exists in the following commands:

hostname command on page 8.14
reload command and reboot command on page 8.19
show cpu command on page 8.22
show cpu history command on page 8.26
show exception log command on page 10.36
show file systems command on page 7.34
show memory command on page 8.32
show memory allocations command on page 8.34
show memory history command on page 8.36
show process command on page 8.41
show system command on page 8.45

Note IPv6 is only supported in VCStack configurations when Virtual MAC addressing is enabled.

Command List

clear counter stack

This command clears all VCStack counters for all stack members.

Syntax

```
clear counter stack
```

Mode Privileged Exec

Example To clear all VC Stack counters:

```
awplus# clear counter stack
```

Related Commands show counter stack

debug stack

This command enables the Virtual Chassis Stacking (VCStack) debugging facilities.

Syntax

```
debug stack [link|topology|trace]

no debug stack [link|topology|trace]
```

Parameter	Description
`link`	Stacking neighbor discovery events on stack links.
`topology`	Stacking topology discovery messages.
`trace`	Notable stacking events.

Default Stack trace debugging is enabled.

Mode Privileged Exec and Global Configuration

Usage This command enables the Virtual Chassis Stacking (VCStack) debugging facilities. It can only be entered on the stack master.

The command displays debug information about the stacked devices. If no parameter is specified, all the stack debugging information will be displayed, including link events, topology discovery messages and all notable stacking events. If link parameter is specified, only the link events debugging information will be displayed.

Examples To enable debugging, enter the following command on the stack master:

```
awplus# debug stack
```

To enable link debugging, enter the following command on the stack master:

```
awplus# debug stack link
```

To enable topology discovery debugging, enter the following command on the stack master:

```
awplus# debug stack topology
```

To enable stack trace debugging, enter the following command on the stack master:

```
awplus# debug stack trace
```

Related Commands undebug stack

license member

This command enables a licensed software feature set on either a specific stack member, or on all stack members.

Use the **no** variant of this command to disable a licensed software feature set on either a specific stack member, or on all stack members provided the specified license has the same set of features on each stack member.

For feature licenses, contact your authorized distributor or reseller. If a license key expires or a proper key is not installed, some software features will not be available.

Note See the AlliedWare Plus^{TM} datasheet for a list of current feature licenses available by product, and the AlliedWare Plus^{TM} How To notes for information on obtaining them.

Syntax

```
license <name> <key> member [<1-8>|all]

no license <name> member [<1-8>|all]
```

Parameter	Description
<*name*>	A unique user-defined name for the license. To determine names already in use, use the show license member command.
<*key*>	The encrypted license key to enable this set of software features.
<1-8>	The ID of the stack member on which the license is to be installed.
all	All stack members.

Mode Privileged Exec

Usage When a licensed software feature is not enabled on all devices within a stack it will result in a mismatch between licenses across the stack, possibly resulting in the stack failing to operate correctly. A warning message is generated if you only specify a single stack member when operating in a stacked configuration:

```
% Warning: licensed features do not match on all stack members.
To prevent the risk of an outage, please resolve.
```

You can obtain a license key that includes a limit on the number of switches that it can be applied to. If the license is applied to a stack that has more members than the license is valid for a warning message is generated and the event is logged.

Examples To enable the license `name1` with the key `12345678ABCDE123456789ABCDE` on all stack members, use the command:

```
awplus# license name1 12345678ABCDE123456789ABCDE member all
```

To remove the license `name1` from all stack members, use the command:

```
awplus# no license name1 member all
```

Validation Command
show license member

Related Commands
license
show license

reboot rolling

This command reboots a stack in a rolling sequence to minimize downtime.

The stack master is rebooted, causing the remaining stack members to failover and elect a new master. The rebooted unit remains separate from the remaining stack and boots up as a stand-alone unit. Once the rebooted unit has finished running its configuration and has brought its ports up, it reboots the remaining stack members in the order of their stack member IDs.

Syntax

```
reboot rolling
```

Mode Privileged Exec

Usage If you are upgrading to a new software version, the new version must also support rollling reboot.

Note When VCStack is used with EPSR, the EPSR **failovertime** must be set to at least 5 seconds. see the epsr command on page 57.4.

Examples To rolling reboot the stack, use the following commands:

```
          awplus# reboot rolling
 Continue the rolling reboot of the stack? (y/n):
```

After running this command, the stack master will reboot immediately with the configuration file settings. The remaining stack members will then reboot once the master has finished re-configuring.

```
 Continue the rolling reboot of the stack? (y/n):
          awplus# y
```

Related Commands boot system
epsr

reload rolling

This command performs the same function as the reboot rolling command on page 78.7.

remote-command <1-8> clear counter stack

This command executes a host-directed clear counter stack command on the specified stack members.

Syntax

```
remote-command <1-8> clear counter stack
```

Parameter	Description
<1-8>	The ID of the stack member where the command should be executed on.
`clear counter stack`	Can be used on a VCStack master to remotely run the clear counter stack command on page 78.3.

Default None

Mode Privileged Exec, and can be also be used in the same mode as the equivalent non-remote command.

Usage This command is used only from the master to execute a command subset that is specific to stack members. If the member ID is not used by any current stack member, the command will be rejected.

Example To clear the stack counters on stack member 2, use the command:

```
awplus# remote-command 2 clear counter stack
```

remote-command <1-4> show

This command enables you to initiate the following show commands on the stack master, but run them on the selected stack member.

Syntax

```
remote-command <1-8>
   show {cpu|counter|exception|file|memory|process|stack|system}
```

Parameter	Description
<1-4>	The ID of the stack member where the command should be executed on.
`counter`	Runs the show counter stack command on page 78.12 for the selected stack member.
`exception`	Runs the show exception log command on page 10.36 for the selected stack member.
`file`	Runs the show file command on page 7.33.
`process`	Runs the show process command on page 8.41 for the selected stack member.
`stack`	Runs the show stack command on page 78.21 for the selected stack member.
`system`	Runs the show system command on page 8.45 for the selected stack member.
`show CPU sort pri`	Runs the show cpu command on page 8.22 to run with the parameters **sort** and **pri** for the selected stack member.
`show CPU sort runtime`	Runs the show cpu command on page 8.22 to run with the parameters **sort** and **runtime** selected for the selected stack member.
`show CPU sort sleep`	Runs the show cpu command on page 8.22 to run with the parameters **sort** and **pri** for the selected stack member.
`show CPU sort thrds`	Runs the show cpu command on page 8.22 to run with the parameters **sort** and **thrds** for the selected stack member.
`show exception log`	Runs the show exception log command on page 10.36 for the selected stack member.
`show file systems`	Runs the show file systems command on page 7.34 for the selected stack member.
`show log permanent`	Runs the show file systems command on page 7.34 for the selected stack member.
`show log permanent`	Runs the show log permanent command on page 10.42 for the selected stack member
`show log permanent tail`	Runs the tail parameter of the show log permanent command on page 10.42 for the selected stack member.

Parameter(cont.)	Description(cont.)
`show memory`	Runs the show memory command on page 8.32 for the selected stack member.
`show memory history`	Runs the **history** parameter of the show memory command on page 8.32 for the selected stack member.
`show memory sort peak`	Runs the **sort** and **peak** parameters of the show memory command on page 8.32 for the selected stack member.
`show memory sort size`	Runs the **sort** and **size** parameters of the show memory command on page 8.32 for the selected stack member.
`show memory sort stk`	Runs the **sort** and **stk** parameters of the show memory command on page 8.32 for the selected stack member.
`show process sort cpu`	Runs the **sort** and **cpu** parameters of the show process command on page 8.41 for the selected stack member.
`show process sort mem`	Runs the **sort** and **mem** parameters of the show process command on page 8.41 for the selected stack member.
`show stack`	Runs the show stack command on page 78.21 for the selected stack member.
`show stack detail`	Runs the **details** parameter of the show stack command on page 78.21 for the selected stack member for the selected stack member.
`show system`	Runs the **details** parameter of the show stack command on page 78.21 for the selected stack member.
`show system environment`	Runs the show system command on page 8.45 for the selected stack member.
`show system pluggable`	Runs the **pluggable** parameter of the show system command on page 8.45 for the selected stack member.
`show system serialnumber`	Runs the **serialnumber** parameter of the show system command on page 8.45 for the selected stack member.

Default None

Mode Privileged Exec, and can be also be used in the same mode as the equivalent non-remote command.

Usage This command is used only from the master to execute a command subset that is specific to stack members. If the member ID is not used by any current stack member, the command will be rejected.

Example To execute the **show system** command on stack member 2, use the command:

```
awplus# remote-command 2 show system
```

remote-login

This command is used only on the master in order to log onto the CLI of another stack member. In most respects the result of this similar to being logged into the stack master. Configuration commands are still applied to all stack members, but show commands, and commands that access the file system are executed locally.

The specific output obtained will vary greatly depending on the show command chosen.

Syntax `remote-login (<1-8>`

Parameter	Description
<1-8>	The specific stack member whose CLI is to be accessed.

Mode Privileged Exec

Usage Note that some commands such as **ping** or **telnet** are not available when the remote-login is used.

Example To log onto stack member 2, use the following command:

```
awplus# remote-login 2
```

To return to the command prompt on the master stack member, type **exit**.

show counter stack

Use this command to display Virtual Chassis Stack (VCStack) related counter information.

Syntax `show counter stack`

Default All counters are reset when the stack member is rebooted.

Mode User Exec and Privileged Exec

Usage If this command is entered on the stack master, it will display all the stacking counter information for every stack member.

When used as a host-directed command, it will display only the stacking counter information for the specific stack member.

Examples To display the stacking counter information about the whole stack, use the following command on the stack master.

```
awplus# show counter stack
```

To display the stacking counter information about stack member 2, use the command:.

```
awplus# remote-command 2 show counter stack
```

Figure 78-1: Example output from the **show counter stack** command

```
Virtual Chassis Stacking counters

Stack member 1:

Topology Event counters
Units joined           ......... 1
Units left             ......... 0
Links up               ......... 1
Links down             ......... 0
ID conflict            ......... 0
Master conflict        ......... 0
Master failover        ......... 0
Master elected         ......... 1
Master discovered      ......... 0
SW autoupgrades        ......... 0

Stack Port 1 Topology Event counters
Link up                ......... 3
Link down              ......... 2
Nbr re-init            ......... 0
Nbr incompatible       ......... 0
Nbr 2way comms         ......... 1
Nbr full comms         ......... 1
```

Figure 78-1: Example output from the **show counter stack** command(cont.)

```
Stack Port 2 Topology Event counters
Link up                   ......... 0
Link down                 ......... 0
Nbr re-init               ......... 0
Nbr incompatible          ......... 0
Nbr 2way comms            ......... 0
Nbr full comms            ......... 0

Topology Message counters
Tx Total                  ......... 4
Tx Hellos                 ......... 4
Tx Topo DB                ......... 0
Tx Topo update            ......... 0
Tx Link event             ......... 0
Tx Reinitialise           ......... 0
Tx Port 1                 ......... 4
Tx Port 2                 ......... 0
Tx 1-hop transport        ......... 4
Tx Layer-2 transport      ......... 0
Rx Total                  ......... 1
Rx Hellos                 ......... 1
Rx Topo DB                ......... 0
Rx Topo update            ......... 0
Rx Link event             ......... 0
Rx Reinitialise           ......... 0
Rx Port 1                 ......... 1
Rx Port 2                 ......... 0
Rx 1-hop transport        ......... 1
Rx Layer-2 transport      ......... 0

Topology Error counters
Version unsupported       ......... 0
Product unsupported       ......... 0
XEM unsupported           ......... 0
Too many units            ......... 0
Invalid messages          ......... 0



Resiliency Link counters
Health status good        ......... 1
Health status bad         ......... 0
Tx                        ......... 0
Tx Error                  ......... 0
Rx                        ...... 3600
Rx Error                  ......... 0

Stack member 2:

-- Output repeated for other stack members - details not shown--
```

Table 78-1: Parameters in the output of the **show counter stack** command

Parameters	**Description**
`Topology Event Counters`	
`Units joined`	Number of times that the stack acquires a member.
`Units left`	Number of times that the stack loses a member.
`Links up`	Number of times that a stack link is up in the stack.
`Links down`	Number of times that a stack link is down in the stack.
`ID conflict`	Number of times that stack member ID conflicts.
`Master conflict`	Number of times that stack master conflict occurs.
`Master failover`	Number of times that stack master fails.

Table 78-1: Parameters in the output of the **show counter stack** command(cont.)

Parameters	Description
`Master elected`	Number of times that stack master is elected.
`Master discovered`	Number of times that stack master is discovered.
`SW autoupgrades`	Number of times that the software in the stack members are auto upgraded.
`Stack port counters`	
`Link up`	Number of times that this unit's physical stack link has come up.
`Link down`	Number of times that this unit's physical stack link has come down.
`Nbr re-init`	Number of times that the neighbor is detected as having reinitialised.
`Nbr incompatible`	Number of times that the neighbor is detected as incompatible.
`Nbr 2way comms`	Number of times that the neighbor is in two way communication. status.
`Nbr full comms`	Number of times that the neighbor is in full communication status.
`Topology message counters`	
`Total`	Number of total topology messages.
`Hellos`	Number of hello messages.
`Topology DB`	Number of topology database messages.
`Topology update`	Number of topology database update messages.
`Link event`	Number of link events messages.
`Reinitialise`	Number of reinitialise messages.
`1-hop transport`	Number of 1-hop transport messages.
`Layer-2 transport`	Number of layer 2 transport messages.
`Link event`	Number of link events messages.
`Reinitialise`	Number of reinitialise messages.
`1-hop transport`	Number of 1-hop transport messages.
`Layer-2 transport`	Number of layer 2 transport messages.
`Topology error counters`	Reasons why a neighboring unit could not join the stack.
`Version unsupported`	Number of stack software version unsupported errors.
`Product unsupported`	Number of Product unsupported errors.
`XEM unsupported`	Number of XEM unsupported errors.
`Too many units`	Number of too may units errors.

Table 78-1: Parameters in the output of the **show counter stack** command(cont.)

Parameters	Description
`Invalid messages`	Number of invalid messages.
`Health status good`	The number of times that the resiliency link has successfully carried healthchecks following a failure at startup.
`Health status bad`	The number of times that the resiliency link healthcheck has timed out. A timeout occurs when a backup stack member detects a delay greater than two seconds between healthcheck messages received.
`Rx`	The total number of healthcheck messages that a stack member has received from the stack master.
`Rx Error`	The total number of invalid healthcheck messages that have been received from the master. This message is not applicable to the stack master.

Related Commands show stack

show debugging stack

This command shows which debugging modes are currently enabled for virtual chassis stacking.

Syntax

```
show debugging stack
```

Mode User Exec and Privileged Exec

Usage To display the stack debugging mode status, use the command:

```
awplus# show debugging stack
```

Figure 78-2: Example output from the show debugging stack command

```
Virtual Chassis Stacking debugging status:
  VCS link debugging is on
  VCS topology debugging is on
  VCS trace debugging is on
```

Related Commands

debug stack
remote-command <1-4> show

show license member

Use this command to display information about either a specific software license, or all software feature licenses enabled on either a specific stack member or all stack members.

Syntax `show license [<name>] [brief] member [<1-8>|all]`

Parameter	Description
<name>	The name of the license to show information about.
`brief`	Display a brief summary of license information.
<1-8>	The ID of the stack member to show information about.
`all`	Display information about all stack members.

Mode User Exec and Privileged Exec

Examples To display a brief summary of information about all enabled licenses on stack member 2, use the command:

```
awplus# show license brief member 2
```

To display full information about all enabled licenses on all stack members, use the command:

```
awplus# show license member all
```

To display full information about the license `name1` on all stack members, use the command:

```
awplus# show license name1 member all
```

Output Figure 78-3: Example output from the **show license member** command

```
awplus#show license index member 1
Stack Member 1
OEM Territory: ATKK
Software Feature Licenses
-------------------------------------------------------------
Index                          : 0
License name                   : Base License
Customer name                  : Base License
Quantity of licenses           : 1
Type of license                : Full
License issue date             : 07-Jul-2000
License expiry date            : N/A
Features included              : VRRP
```

Table 78-2: Parameters in the output of the **show license member** command

Parameter	Description
`Index`	Index identifying entry.
`License name`	Name of the license key bundle (case-sensitive).
`Customer name`	Customer name.
`Quantity of licenses`	Quantity of licensed installations.
`Type of license`	Full or Temporary.
`License issue date`	Date the license was generated.
`License expiry date`	Expiry date for temporary license.
`Features included`	List of features included in the license.

Figure 78-4: Example output from the **show license brief member <*1-8*>** command

```
awplus#show license brief member 2
Stack Member 2
OEM Territory: ATKK
Software Feature Licenses
--------------------------------------------------------------
Index License name    Quantity     Customer name
      Type                         Period
--------------------------------------------------------------
0     Base License    1            Base License
      Full                         N/A

Current enabled features for displayed licenses:
 , VRRP
```

Table 78-3: Parameters in the output of the **show license brief member** command

Parameter	Description
`Index`	Index identifying entry.
`License name`	Name of the license key bundle (case-sensitive).
`Quantity`	Quantity of licensed installations.
`Customer name`	Customer name.
`Type`	Full or Temporary.
`Period`	Expiry date for temporary license.
`Current enabled features for displayed licenses`	List of features included in the license.

Related Commands

license
license member
show license

show running-config stack

Use this command to display the running system information specific to the virtual chassis stack.

```
show running-config stack
```

Mode Privileged Exec and Global Configuration

Example To display the stacking running configuration information, use the command:

```
awplus# show running-config stack
```

Output Figure 78-5: Example output from the **show running-config stack** command

```
stack management vlan 4000
stack management subnet 192.168.0.0
no stack 1 software-auto-synchronize
no stack 4 software-auto-synchronize
stack 2 priority 0
```

Related Commands remote-command <1-4> show

show running-config

show provisioning (stack-member)

Use this command to display the provisioning status of all installed or provisioned hardware. Provisioning is the preconfiguration necessary to accommodate future connection of hardware items such as a switch.

Syntax `show provisioning`

Mode User Exec and Privileged Exec

Example To show provisioning, use the following command:

```
awplus# show provisioning
```

Output Figure 78-6: Example output from the **show provisioning** command

```
Switch provisioning summary information

ID  Board class Status
1.0 x510-28     Hardware present
2.0 x510-28     Hardware present
3.0 x510-52     Hardware present
4.0 x510-52     Provisioned
```

Table 78-4: Parameters in the output of the **show provisioning** command

Parameter	Description
`ID`	The unit.bay-location of the provisioned hardware.
`Board class`	The hardware type.
`Status`	The provisioned state: ■ Hardware Present means that the hardware is currently installed in the stack. ■ Provisioned means that although the hardware is not currently installed; the stack is preconfigured ready to accept the hardware installation.

Related Commands show stack
switch provision

show stack

Use this command to display information about current stack members.

Syntax `show stack [detail]`

Parameter	Description
`detail`	Display detailed stacking information.

Default Display summary information only.

Mode User Exec and Privileged Exec

Usage This command displays information about current stack members. If the **detail** parameter is specified, additional information will be displayed for each stack member. By default, only summary information is displayed.

This command can be entered on any stack member as a host-directed command. However, all stack members display the same stacking information.

Examples To display basic information about the stack, use the command:

```
awplus# show stack
```

To display the detailed stacking information about the whole stack:

```
awplus# show stack detail
```

Output Figure 78-7: Example output from the **show stack** command

```
Virtual Chassis Stacking summary information

ID  Pending ID  MAC address       Priority  Status  Role
1   -           0000.cd28.07e1    128       Ready   Active Master
2   -           0015.77c2.4d44    128       Ready   Backup Member
3   -           0015.77c9.7464    128       Syncing Backup Member
4   -           -                 -         -       Provisioned

Operational Status                 Normal operation
Stack MAC address                  0000.cd28.07e1
```

Table 78-5: Parameters in the output from the **show stack** command

Parameter	Description
`ID`	Stack member ID
`MAC address`	Stack member MAC address
`Priority`	Stack member master election priority (between 0 and 255) Note that the lowest number has the highest priority.
`Role`	Stack member's role in the stack, this can be one of: ■ **Active Master**. ■ **Disabled Master** - The temporary master when there is a communication break within the stack, but communication (with the stack master) still exists across the resiliency link. In this state all switch ports within the stack are disabled by default, but a different configuration can be run by a "type stack disabled-master" trigger). ■ **Backup Member** - An ordinary member of the stack. ■ **Provisioned** - Indicates that the stack position is provisionally configured, i.e. ready to accept a particular switch type into the stack.

Table 78-6: Parameters in the output from the **show stack detail** command

Parameter	Description
`Auto upgrade`	Whether the software-auto-configuration feature is turned on, or off.
`Host name`	The host name of the stack member.
`ID`	The stack member ID.
`Last Role Change`	The date and time with the stack member last changed its role in the stack.
`MAC address`	Stack member MAC address.
`Management VLAN ID`	The VLAN ID currently used for stack management: Default is 4094.
`Management VLAN subnet address`	The current stacking management VLAN subnet address.
`Virtual Chassis ID`	The Virtual Chassis ID determines the last 12 bits of the Virtual MAC address: `0000.cd37.0xxx`
`Virtual MAC Address`	The Virtual MAC address of the stack.
`Disabled Master Monitoring`	The current Disabled Master Monitoring status. This can be: ■ Enabled ■ Disabled ■ Inactive
`Operational Status`	The status of the stack - either **enabled** or **disabled.**
`Stack Status`	The stack's overall status. Note that a warning is issued if the stack is not connected in a standard ring topology.
`Pending ID`	The pending stack member ID. This can be changed by the stack renumber command on page 78.32. If there is no pending ID, the "–" symbol will display.
`Stack port status`	The status of the stack port. This can be: ■ **Down** - neighbor incompatible ■ **Discovering neighbor**, or **Learnt neighbor <*neighbor member = ""id>=""*>**
`Priority`	Stack member master election priority (between 1 and 255) Note that the lowest number has the highest priority.
`Product Type`	Stack member product type. For examplex510-28GTX.
`Provisioned`	Indicates that the stack position is provisionally configured, i.e. ready to accept a particular switch type into the stack.

Table 78-6: Parameters in the output from the **show stack detail** command(cont.)

Parameter	Description
`Resiliency link status`	The current status of the resiliency link. The status can be one of: ■ **Not configured**, (Master or Member). ■ **Configured** (Master only). ■ **Successful**: Successfully receiving healthchecks from the Active Master. ■ **Failed** (Member only): Not receiving any healthchecks from the Active Master. ■ **Stopped**: The resiliency link is configured, but is inactive. This may occur in a Disabled Master stack, for example if the Disabled Master Monitoring feature is not used.
`Role`	Stack member's role in the stack, this can be one of: ■ **Active Master**. ■ **Disabled Master** (The temporary master when there is a communication break within the stack, but communication still exists across the resiliency link. In this state all switch ports within the stack are disabled by default, but a different configuration can be run by a "type stack disabled-master" trigger command). ■ **Backup Member** (a device other than the stack master). ■ **Discovering** - joining the stack.
`Status`	Indicates how readily a stack member can take over as master if the current stack master were to fail. ■ **Init** - the stack member is completing the startup initialization. ■ **Syncing** - the stack member is synchronizing state information with the stack master following startup. ■ **Ready** - the stack member is fully synchronized with the current master and is ready to take over immediately.

Related Commands

show counter stack
show stack resiliencylink
stack disabled-master-monitoring
stack resiliencylink
stack software-auto-synchronize

show stack resiliencylink

Use this command to display information about the current status of the resiliency-link across the members of the stack.

```
show stack resiliencylink
```

Mode User Exec and Privileged Exec

Example To display information about the current status of the resiliency-link across the stack members, use the command:

```
awplus# show stack resiliencylink
```

Output Figure 78-8: Example output from the **show stack resiliencylink** command

```
awplus(config)# show stack resiliencylink
Stack member 1:
-----------------------------------------------------------
Status                             Configured
Interface                          vlan4093
Interface state                    UP
Resiliency-link port(s)            port1.2.11

Stack member 2:
-----------------------------------------------------------
Status                             Successful
Interface                          vlan4093
Interface state                    UP
Resiliency-link port(s)            port2.2.11
```

Table 78-7: Parameters in the output of the **show stack resiliencylink** command

Parameter	Description
`Status`	The current status of the stack member's resiliency link. Can be one of: ■ **Not configured**, (Master or Member). ■ **Configured** (Master only). ■ **Successful**: Successfully receiving healthchecks from the Active Master. ■ **Failed** (Member only): Not receiving any healthchecks from the Active Master. ■ **Stopped**: The resiliency link is configured, but is inactive. This may occur in a Disabled Master stack, for example if the Disabled Master Monitoring feature is not used.
`Interface`	The name of the VLAN interface that is connected to the resiliency link.
`Interface state`	The current status of the interface. Can be one of: ■ Up ■ Down
`Resiliency-link port(s)`	The switch port(s) the resiliency link is connected to.

See Also show stack detail
stack resiliencylink
switchport resiliencylink

stack disabled-master-monitoring

This command enables the Disabled Master Monitoring (DMM) feature. If a stack member becomes a disabled master, the DMM feature will use the stack resiliency link to continue monitoring the health of the separated stack master.

Use the **no** variant of this command to disable the DMM feature.

Syntax

```
stack disabled-master-monitoring

no stack disabled-master-monitoring
```

Default By default, Disabled Master Monitoring is enabled. However, it only operates if there is a resiliency link.

Mode Global Configuration

Usage This command enables additional stack resiliency link functionality, which is used if a stack separation occurs. For DDM to operate, a resiliency link must also be configured (stack resiliencylink command on page 78.34). A stack separation could result in a stack member becoming a disabled master, which has the configuration as a normal stack master except that all its switchports are shutdown.

For more information about the disabled master state, see “Disabled Master” on page 77.16.

When the DMM feature is enabled, the disabled master will continue to monitor the health of the original stack master over the stack resiliency link connection. If the original stack master were to fail, when the DMM feature is enabled, then the disabled master will detect this and will automatically re-enable its switchports. This ensures that the stack will continue to pass network traffic, even if a catastrophic stack failure occurs.

For more information about the DMM feature when the stack member is a disabled master, see “Disabled Master Monitoring (DMM)” on page 77.17.

Examples To enable the DMM feature, use the following commands:

```
        awplus# configure terminal

awplus(config)# stack disabled-master-monitoring
```

To disable the DMM feature, use the following commands:

```
        awplus# configure terminal

awplus(config)# no stack disabled-master-monitoring
```

Related Commands

show stack
stack resiliencylink
type stack disabled-master
type stack master-fail

stack enable

This command is used on a stackable stand-alone switch to manually turn on the virtual chassis stacking feature and XEM-STK links.

By default, the VCStack feature starts automatically at the device start-up when XEM-STK is detected.

This command is run on a switch that has previously been removed from the stack (by using the **no** variant of this command) in order to return its stack membership member.

The **no** variant of this command will remove a selected stack member switch from the virtual chassis stack.

Note IPv6 is only supported on VCStacks if virtual MAC addressing is enabled.

Syntax

```
stack enable

no stack <1-8> enable
```

Mode Global Configuration

Usage Running the **no** variant of this command will remove the selected stack member from the VCStack. At this point the removed member will act as a stand-alone master and will disable all of its ports. The switch can then only be accessed via its console port.

To return the switch to stack membership, you first run the **stack enable** command. Note that to do this you must direct connectivity via the console port. Then you must run the reboot command on page 8.19. This will reboot the switch and it will re-join the stack as an ordinary member.

Note the following conditions of applying the **no stack <*1-8*> enable** command:

- If the specified member is a stack master, this command will be rejected.
- If the specified member ID is not used by any current stack member, the command will be rejected.

Caution **Disabling a stack member can significantly degrade the throughput capability of the stack.**

Example To turn on stacking on a stackable stand-alone unit, use the command:

```
        awplus# configure terminal
awplus(config)# stack enable
```

Related Commands reboot

stack management subnet

This command configures the stack's VLAN subnet management address.

Use the **no** variant of this command to reset the stack's VLAN subnet management address back to the default address and mask (192.168.255.0/28).

Syntax

```
stack management subnet <ip-address>
```

```
no stack management subnet
```

Parameter	Description
`<ip-address>`	The new subnet address for VCStack management VLAN.

Default The default stacking management VLAN subnet address is 192.168.255.0 with a subnet mask 255.255.255.240 or /28.

Mode Global Configuration

Usage This command is used only in the master and configures the stack management VLAN subnet address.

The management VLAN will be used for high speed communication between stacked units via the inter-stack connection. Although this command enables you to change the IP address command, the subnet mask must always remain as shown.

The VCStack management IP subnet is solely used internally to the stacked devices, and cannot be reached external to the stack. You should only change the VCStack management VLAN subnet address if it causes a conflict within your network.

Note that several separate stacks can use the same default management VLAN subnet address even though their user ports may share the same external network. If the VCStack subnet address is changed, then the configuration for any new units must also be updated before they are inserted into the stack.

If the management VLAN subnet address is changed by this command, you can use the **no** variant of this command reset it to its default.

Example To set the management VLAN subnet address to 192.168.255.144:

```
        awplus# configure terminal
awplus(config)# stack management subnet 192.168.255.144
```

Related Commands stack management vlan

stack management vlan

Use this command to configure the virtual stack management VLAN ID. It can only be entered from the stack master.

Use the **no** variant of this command to change the virtual stack management VLAN ID back to the default (VLAN ID 4094).

Syntax

```
stack management vlan <2-4094>
```

```
no stack management vlan
```

Parameter	Description
<2-4094>	VCStack management VLAN ID. The default is 4094.

Default The default VCStack management VLAN ID is 4094.

Mode Global Configuration

Usage This command is used only in the master and configures the VCStack management VLAN ID.

The management VLAN is used for high speed communication between stacked units via the XEM-STK or the two back panel stacking ports (SwitchBlade® x908 only). This command enables you to change the ID of this VLAN.

The default stacking management VLAN ID is 4094, which is the last configurable VLAN ID in the switch.

The VCStack management VLAN is created and configured automatically so that the VCStack VLAN cannot be used in the stack's VLAN configuration commands (such as `awplus(config-vlan)# vlan <VCS management VLAN ID>`).

The management VLAN should only be changed if the default VCStack VLAN ID needs to be used in the stack's VLAN configuration.

Caution

When the command is entered, the updated management VLAN configuration will take effect once the stack is restarted.

If the management VLAN ID is changed by this command, you can use the no variant of this command to change it back to default value.

Examples To set the management VLAN to 4000, enter the following commands:

```
        awplus#  configure terminal
awplus(config)#  stack management vlan 4000
```

To reset the management VLAN back to the default (4094), enter the following commands:

```
        awplus#  configure terminal
awplus(config)#  no stack management vlan
```

Related Commands stack management subnet

stack priority

Use this command to changes a specific stack member's stack ID and its master-election priority.

Syntax

```
stack <1-8> priority <0-255>

no stack <1-8> priority
```

Parameter	Description
stack <1-8>	Sets the ID of the stack member being configured.
priority *<0-255>*	Sets a numeric value for the stack member's priority to become stack master. The lowest numeric value is assigned the highest priority. The default is 128.

Mode Global Configuration

Usage This command is used to change the value of a specific stack member's master-election priority. If the specified member ID is not used by any current stack member, the command will be rejected.

The election criteria selects the stack member whose priority numeric has the **lowest value** to become the stack master. Where two stack members both have the same lowest priority value, then the stack member with the lowest MAC address will be elected as master.

Note Assigning a new priority value will not immediately change the current stack master. In order to force a master re-election after the new priority value is assigned, use reboot stack-member <master's ID> to reboot the current stack master, a new stack master will then be elected based on the new priority values.

Example To change the priority of stack member 4 to be 2, use the commands:

```
        awplus#  configure terminal
awplus(config)#  stack 4 priority 2
```

Validation Command show stack

stack renumber

Use this command to change the ID of a specific stack member.

Syntax

```
stack <member's-existingID> renumber <member's-newID>
```

Parameter	Description
`<member's-existingID>`	The the stack member ID to be renumbered - also referred to as the "current member ID". Can have a value in the range 1 to 8.
`renumber`	Change the existing stack member ID.
`<member's-newID>`	The stack member's new ID - also referred to as the new member ID. Can have a value in the range 1 to 8. Note that when operating two member stacks, we advise using stack IDs 1 and 2.

Default Every stack unit will initially try to use stack member ID of 1.

Mode Global Configuration

Usage This command is used to change the ID of a specific stack member - primarily when exchanging stack members. The changes made by this command will not take effect until the switch is rebooted.

Note This command does not alter any of the stacks's existing configuration, apart from the member-ID specified. For example, if stack member 2 were removed from the stack and a new stack unit is assigned the member 2 stack ID then the interface configuration that existed for the removed stack member 2 (including its priority number) will be applied to the new stack member 2 when the switch is rebooted.

The ***current member ID*** must already be assigned to an existing stack member. To avoid duplicating IDs, a warning message will appear if you assign a ***new member ID*** that is currently assigned to another stack member. However, you can continue to rename the stack member IDs and remove ID duplications. If you do not remove the duplications, then the device with the highest root priority will be allocated this ID. Once you have removed any duplicate IDs, you can reboot the switch to implement your changes.

Note that the configured member-ID is saved immediately on the renumbered member, and so is not reliant on using the copy running-config command for it to take effect

Example To change the stack member ID 2 to be member ID 3, use the commands:

```
        awplus# configure terminal
awplus(config)# stack 2 renumber 3
```

Validation Command show stack

stack renumber cascade

This command is used to renumber the members of a stack so that their IDs are ordered sequentially, relative to the member's physical position within the stack.

Caution Changing the stack numbering will upset the existing stack member configurations such as port settings etc. This command is intended for use when the stack is either initially commissioned, or has undergone a major reconfiguration. In this situation you run the stack renumber command (which will automatically reboot the switch), then configure the stack members to meet the new requirements.

Syntax

```
stack <1-8> renumber cascade [<1-8>]
```

Parameter	Description
`<1-8>`	The ID of the stack member to start renumbering from.
`renumber`	Change the existing stack member ID.
`cascade`	Renumber the existing stack member ID in cascade order.
`<1-8>`	The new ID for the first member renumbered.

Default If no member-ID is specified, the member will take the default ID of 1.

Mode Global Configuration

Usage This command is used to renumber the members of a stack so that their member's IDs are ordered sequentially, based on physical order of the XEM-STK connections. This would normally be done either when the stack is initially configured or following a major reconfiguration.

This command is equivalent to pressing and holding the select button on the XEM-STK to renumber the stack members. The renumber will start on the specified stack member. If that member ID is not used by any of the existing stack member, the command will be rejected.

The starting stack member will be renumbered with the new member-ID specified, or the default of member ID of 1. The stack ID of the next physically will be the starting members ID +1, for example member ID 2. This renumbering will continue in cascading order around the stack members.

The changes will take place immediately and reboot all stack members. For this reason a confirmation prompt follows this command entry, asking whether you are sure you want to renumber and reboot the entire stack.

Example

```
awplus(config)# stack 1 renumber cascade

Any existing interface configuration may no longer be valid

Are you sure you want to renumber and reboot the entire stack?
(y/n): y
```

Related Commands show stack
stack renumber

stack resiliencylink

This command configures the resiliency link used by Virtual Chassis Stacking. The interface used is a dedicated VLAN (resiliencylink VLAN) to which switch ports may become members. This VLAN is dedicated to the resiliency link function and must not be the stack management VLAN.

Syntax

```
stack resiliencylink <interface>
```

```
no stack resiliencylink
```

Mode Global Configuration

Usage The resiliency-link is only used when a backup member loses connectivity with the master via the stacking cables. Such a communication loss would occur if:

- a XEM-STK is removed or fails,
- two or more XEM-STK cables are unplugged or fail,
- the stack master itself fails due to a reboot or power failure.

The resiliency-link allows the backup member to determine if the master is still present in the network by the reception of health-check messages sent by the master over the resiliency-link interface.

Reply health-check messages are received if the master is still online, but the stack will now split into two different 'stubs'. The stub containing the existing master will continue operating as normal. The members in the masterless stub will now use a "type stack disabled-master" trigger to run a configuration to form a second temporary stack. This utilizes the remaining stack members' resources without conflicting directly with the master's configuration. If no "type stack disabled-master" trigger was configured on the switches, then the masterless stub members will disable their switch ports.

If no health-check messages are received, then the master is assumed to be completely offline, and so the other stack members can safely take over the master's configuration.

Caution ⚠ **The purpose of the resiliency link is to enable the stack members (particularly the backup master) to check the status of the master under fault conditions. If the resiliency link is not configured, and the master loses communication with its other stack members, then Virtual Chassis Stacking will assume the master is NOT present in the network, which could cause network conflicts if the master is still on line. Note that this is a change to the stacking of releases prior to version 5.3.1.**

Example To set the resiliency link to be VLAN 4093.

First use the **stack resiliencylink** command to create the resiliency `vlan 4093`

```
        awplus#  configure terminal
awplus(config)#  stack resiliencylink vlan4093
```

Next use the switchport resiliencylink command to assign the resiliencylink vlan to the interface port, in this case `port1.0.1`.

```
          awplus# configure terminal
  awplus(config)# interface port1.0.1
awplus(config-if)# switchport resiliencylink
```

Related Commands

show stack
show stack resiliencylink
stack disabled-master-monitoring
switchport resiliencylink

stack software-auto-synchronize

This command is used only on the stack master and enables the software version auto-synchronization feature either on a specified stack member or all stack members and candidates. A stack candidate is a switch that is about to join a stack.

Use the **no** variant of this command to turn the software version auto synchronization feature off.

Syntax

```
stack {all|<1-8>} software-auto-synchronize

no stack {all|<1-8>} software-auto-synchronize
```

Parameter	Description
`all`	All stack members.
`<1-8>`	The ID of a stack member.

Default All the stack members have the stack software-auto-synchronize feature enabled by default.

Mode Global Configuration

Usage This command is used to enable the software version auto-synchronization feature for either a specific stack member or all stack members and candidates.

Note that if a stack candidate attempts to join a stack, but is running a software release that is different to the other stack members, then the software version auto-synchronization feature will copy the master's software release onto the new candidate. If the software version auto-synchronization feature is not enabled, then the candidate will be rejected from the stack.

Note that the software version auto-synchronization feature may also result in the stack candidate downgrading its software release if the master is running an older software version.

Examples To turn on the software-auto-synchronize feature on stack member 2, which was previously turned off, use the following commands:

```
        awplus#  configure terminal
  awplus(config#  stack 2 software-auto-synchronize
```

To turn on the software-auto-synchronize feature for all stack members, which were previously turned off, use the following commands:

```
        awplus#  configure terminal
 awplus(config)#  stack all software-auto-synchronize
```

Validation Command show stack

stack virtual-chassis-id

This command specifies the VCStack virtual chassis ID. The ID selected will determine which virtual MAC address the stack will use. The MAC address assigned to a stack must be unique within its network.

Note The command will not take effect until the switch has been rebooted.

Syntax

```
stack virtual-chassis-id <id>
```

Parameter	Description
<id>	The value of the ID - enter a number in the range 0 to 4095.

Mode Global Configuration

Usage The virtual-chassis-id entered will form the last 12 bits of a pre selected MAC prefix component; that is, `0000.cd37.0xxx`. If you enable the VCStack virtual MAC address feature (by using the stack virtual-mac command) without using the stack virtual-chassis-id command to select the virtual-chassis-id, then the stack will select a virtual-chassis-id from a number within the assigned range.

Example To set the stack virtual-chassis-id to 63 use the commands

```
        awplus# configure terminal
awplus(config)# stack virtual-chassis-id 63
```

This will result in a virtual MAC address of: `0000.cd37.003f`.

Related Commands

show running-config
show stack detail
stack virtual-mac

stack virtual-mac

This command enables the VCStack virtual MAC address feature. For more information on this topic refer to "Fixed or Virtual MAC Addressing" on page 77.12. With this command set, the value applied for the virtual MAC address is determined by the setting of the command stack virtual-chassis-id command on page 78.37.

Note This command will not take effect until the switch has been rebooted.

Syntax

```
stack virtual-mac

no stack virtual mac
```

Mode Global Configuration

Example

```
        awplus# configure terminal
awplus(config)# stack virtual mac
```

Related Commands

show running-config
show stack detail
stack virtual-chassis-id

switch provision

This command enables you provide the configuration for a new VCStack member switch prior to physically connecting it to the stack. To run this command, the stack position must be vacant. The selected hardware type must be compatible existing stack hardware.

Use the **no** variant of this command to remove an existing switch provision.

Syntax

```
switch <1-4> {provision|reprovision}{x510-28|x510-52}

no switch <1-8> provision
```

Parameter	Description
`<1-8>`	The stack member switch position to be provisioned.
`provision`	Provides settings within the stack configuration ready for a specific switch type to become a stack member.
`reprovision`	Reconfigure an existing provision configuration.
`x510-28`	Provision an x510-28 port switch.
`x510-28`	Provision an x510-52 port switch.

Mode Global Configuration

Examples x600: To provision an x510-28 port switch as stack member 2, use the following commands:

```
        awplus# configure terminal
awplus(config)# switch 2 provision x510-28
```

Related Commands show provisioning (stack-member)

switchport resiliencylink

This command configures the switch port to be a member of the stack resiliency link VLAN. Note that this switchport will only be used for stack resiliency-link traffic and will not perform any other function, or carry any other traffic.

The **no** variant of this command removes the switchport from the resiliency link VLAN.

Syntax

```
switchport resiliencylink
```

```
no switchport resiliencylink
```

Mode Global Configuration

Examples To set the resiliency link to be VLAN 4093.

First use the stack resiliencylink command to create the resiliency `vlan4093`

```
         awplus# configure terminal
 awplus(config)# stack resiliencylink vlan4093
```

Next use the switchport resiliencylink command to assign the resiliencylink vlan to the 31

port, in this case `port1.0.1`.

```
            awplus# configure terminal
    awplus(config)# interface port1.0.1
 awplus(config-if)# switchport resiliencylink
```

See Also stack resiliencylink
show stack resiliencylink

undebug stack

This command applies the functionality of the no debug stack command on page 78.4.

Appendix A: Command List

Access Lists

Appendix B Changes in Version 5.4.2A-0.1

Software version 5.4.2A-0.1 is the first release on the x510 series of switches. This appendix therefore, contains no update information.

Appendix C: GUI Reference

Introduction

This appendix describes how to install, configure and use the Graphical User Interface (GUI) on switches running the AlliedWare Plus™ OS. The GUI provides extensive monitoring and essential configuration functionality for Allied Telesis switches via a web browser. This document explains how to install the GUI using either a USB flash drive, or via a TFTP server.

The GUI functionality is provided via a Java applet file. Before you can use the GUI to manage your switches, you must download the Java applet file, and install it to your switch's Flash file system.

Once the Java applet file is present in your switch's Flash, no specific commands are required to enable the GUI, or to inform the switch which Java applet file to use. Instead, when an incoming browser connection is established with the switch, the switch will automatically send the most recent compatible Java applet file that is present in its Flash file system.

Different versions of the Java applet file will be compatible with different versions of the AlliedWare Plus^{TM} OS. The AlliedWare Plus^{TM} OS automatically determines if a Java applet file is compatible, so the Java applet file that is delivered to your browser will always be compatible with the AlliedWare Plus^{TM} OS version running on the switch to which you have connected.

Note which products and software version the GUI works with, along with PC and browser specifications listed. You may need to install and run the latest Java Runtime Environment that you can download from the Sun site so your browser can fully support the GUI Java applet.

Installing the GUI and setting the switch

This section shows you how to install and setup the AlliedWare Plus^{TM} GUI on your switch.

System Requirements

To install and run the AlliedWare Plus^{TM} GUI you will require the following system products and setup:

- PC Platform:
  Windows XP SP2 and up / Windows Vista SP1 and up
- Browser: (must support Java Runtime Environment (JRE) version 6)
  Microsoft Internet Explorer 7.0 and up / Mozilla Firefox 2.0 and up

Installing the GUI to your switch from a USB flash drive

Step 1: Download a GUI Java applet

The GUI Java applet file is available in a compressed (zip) file with the AlliedWare Plus^{TM} Operating System software from the Support area of the Allied Telesis Website: http://www.alliedtelesis.com. Download the Java applet file. This file will have a `.zip` file name extension. You need to extract the Java `.jar` file from the compressed `.zip` file. The version number of the software applet file (`.jar`) gives the earliest version of the software file (`.rel`) that the GUI can operate with.

Step 2: Copy the GUI Java applet .jar file to a USB flash drive

Insert the USB flash drive into the USB socket on the front of your switch. Connect to the management port, then login to the switch.

Copy the GUI Java applet to your switch, using the following command:

```
awplus# copy usb:/<filename.jar> flash:/
```

Where `<filename.jar>` is the GUI Java applet file you downloaded in Step 1.

Note **Where the GUI file is not in the root directory of the USB flash drive, you must enter the full path to the GUI file. For example, where the GUI file resided in the folder gui_files, you would enter the command:**
copy usb:/gui_files/filename.jar flash:/

Installing the GUI to Your Switch Via TFTP server

Step 1: Download a GUI Java applet file from the support site:

The GUI Java applet file is available in a compressed (`.zip`) file with the AlliedWare Plus^{TM} Operating System software from the Support area of the Allied Telesis Website: http://www.alliedtelesis.com. Download the Java applet file. This file will have a `.zip` file name extension. You need to extract the Java `.jar` file from the compressed `.zip` file. The version number of the software applet file (.jar) gives the earliest version of the software file (`.rel`) that the GUI can operate with.

Step 2: Copy the GUI applet

Copy the GUI applet **.jar** file onto a TFTP server. Ensure this TFTP server is enabled and ready for the switch. Connect to the management port of the switch, then login to the switch. Do not connect to the management port of the TFTP server.

Step 3: Copy the GUI Java applet to your switch

Use the following commands to copy the GUI Java applet to your switch:

```
awplus# copy tftp://<server-address>/
        <filename.jar> flash:/
```

Where *<server-address>* is the IP address for the TFTP server, and where *<filename.jar>* is the GUI Java applet file you downloaded in Step 1.

Setting up your switch and logging into the GUI

Step 1: Assign the IP addresses:

Normally the GUI is assigned to VLAN1, because all switch ports are assigned to this VLAN by default. If this VLAN already has an IP address assigned to it, then you can skip the remainder of this step; if no IP address has been assigned to the VLAN, use the following commands to configure an appropriate IP address for the VLAN used for GUI connectivity:

```
          awplus#  configure terminal
  awplus(config)#  interface vlan1
awplus(config-if)# ip address <address>/<prefix-length>
```

Where *<address>* is the IP address that you will subsequently browse to when you connect to the GUI Java applet. For example, to give the switch an IP address of `192.168.2.6`, and a subnet mask of `255.255.255.0`, use the following command:

```
awplus(config-if)# ip address 192.168.2.6/24
```

Step 2: Configure the Default Gateway

In necessary, use the following commands to configure the default gateway.

```
awplus(config-if)# exit
   awplus(config)# ip route 0.0.0.0/0 <gateway address>
```

Where *<gateway-address>* is the IP address for your gateway device. Note that you do not need to define a default gateway if you browse to the switch from within its own subnet.

Step 3: Create a user account

In order to log into the GUI, you must first create a user account. Use the following command to setup a user account:.

```
awplus(config)# username <username> privilege 15
                password <password>
```

Note that you can create multiple users to log into the GUI. See the AlliedWare Plus Software Reference for information about the username command. The switch must be configured with a local database user, or the switch must be configured to remotely authenticate users with either TACACS+ or RADIUS.

Step 4: Ensure HTTP service is enabled

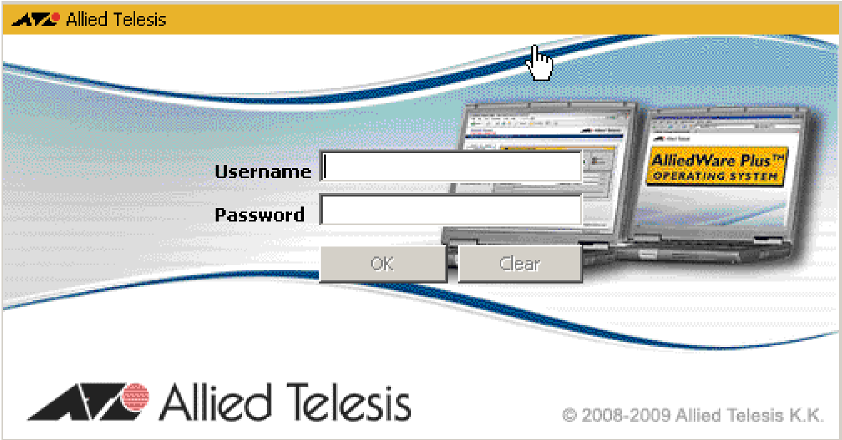
The HTTP service needs to be enabled on the switch before it accepts connections from a web browser. The HTTP service is enabled by default. However, if the HTTP service has been disabled then you must enable the HTTP service again. If the HTTP service is disabled then use the following command to enable the HTTP service:

```
awplus(config)# service http
```

See the AlliedWare Plus^{TM} Software Reference for information about the service http command.

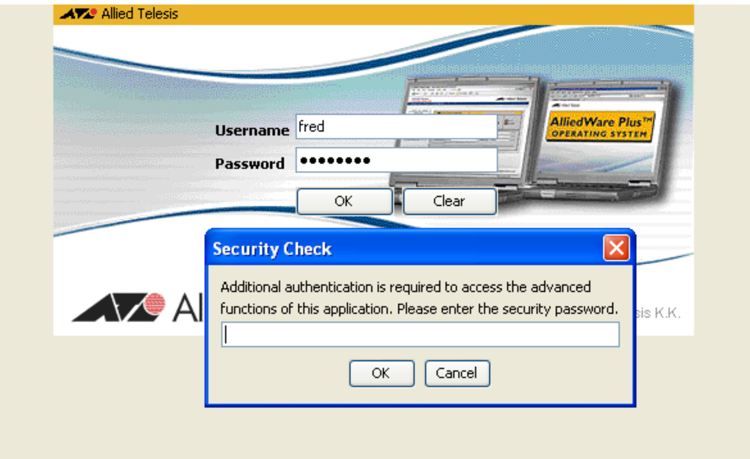
Step 5: Logging into the GUI

Start a browser then enter the IP address you configured in Step 1 as the URL. You will then be presented with a login screen after the GUI Java applet has started. You can then Log in with the username and password that you defined previously in Step 3.



Step 6: Security Check

You may also be presented with a security check password prompt. This will occur when you have logged into the GUI with a user that is configured on the switch and have a privilege level of less than 15, or if the switch has been configured to authenticate enable passwords via TACACS+ using the aaa authentication enable default group tacacs+ command.

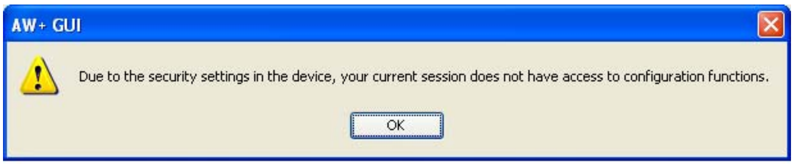


Username: fred

Password: ••••••••

Additional authentication is required to access the advanced functions of this application. Please enter the security password.

You must enter the privilege level 15 enable password configured on the switch to access GUI configuration dialogs. If you enter an incorrect enable password, or no privilege level 15 enable password has been configured, the a message is shown stating you can use the GUI to monitor the switch, but not to configure the switch.



Due to the security settings in the device, your current session does not have access to configuration functions.

Using the GUI

This section explains how to use the AlliedWare Plus^{TM} GUI. It assumes that you have installed the GUI on your switches and have the setup the browser on your PC. This procedure is covered in "Installing the GUI and setting the switch" on page C.4.

In this section each screen is presented by its tab name and explains the content of the screen components.

System > Status

The **System** > **Status** menu tab enables you to display and configure basic system information.

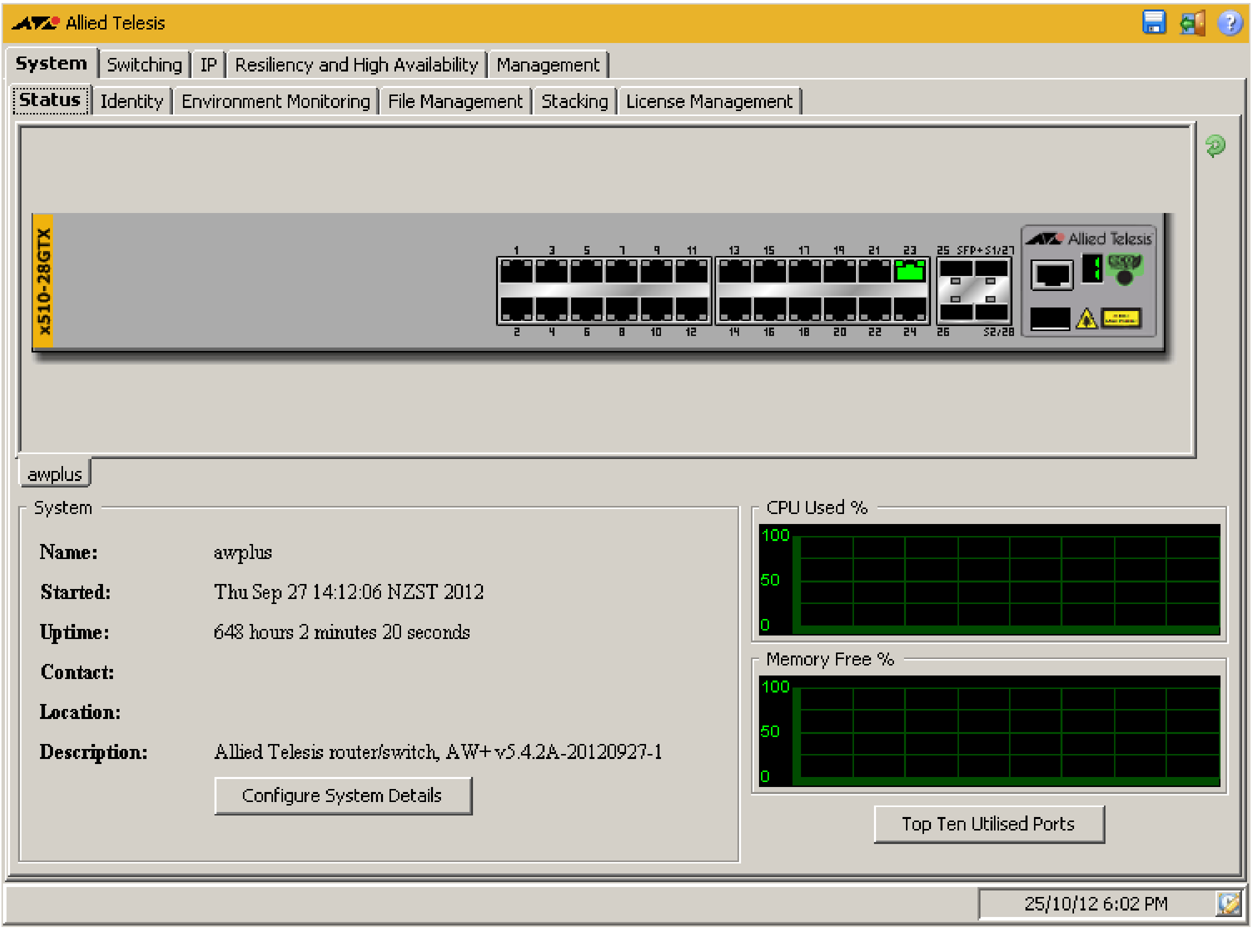
The **CPU Used %** and **Memory Free %** graphs provide a brief history of CPU and memory usage.

Note For systems equipped and configured using VCStack, there is a separate tab for each stack member with the system name displayed on each tab.

The last two SFP+ port LEDs are lit when VCStack is enabled on the switch.

Menu Tab

Figure C-1: Example showing the **System > Status** menu tab:



Name: awplus

Started: Thu Sep 27 14:12:06 NZST 2012

Uptime: 648 hours 2 minutes 20 seconds

Contact:

Location:

Description: Allied Telesis router/switch, AW+ v5.4.2A-20120927-1

25/10/12 6:02 PM

Description

Display Label / Field	Description
`System / Name`	Specifies the network name of the system, as set with the 'hostname' command in the CLI.
`System / Started`	Date and time the switch was last booted.
`System / Uptime`	Elapsed time since the last boot.
`System / Contact`	Contact details for system maintenance.
`System / Location`	Location of the switch
`System / Description`	Description of the switch, including manufacturer, model, and software version.
`Top Ten Utilised Ports`	Displays a sorted list of the ten most used ports listed by port and its utilization. You can rearrange and resort the list by port or utilization.

Description

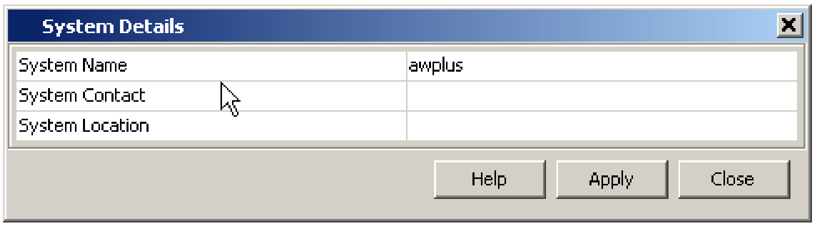
Configuration Button / Field	Description
`System Time & Date (icon)`	Add or modify System Date, System Time, UTC Time Zone Offset.
`Configure System Details`	Add or modify System Name, System Contact, System Location.
`Configure System Details / System Name`	Configures the network name of the system.
`Configure System Details / System Contact`	Configures the contact information for the system, from 0 to 255 characters long. Valid characters are any printable characters and spaces.
`Configure System Details / System Location`	Configures the location of the system, from 0 to 255 characters long. Valid characters are any printable characters and spaces.

System > Status > System Details

The **System > Status > System Details** dialog allows you to configure basic system information.

Configuration Dialog

Figure C-2: Example showing **System > Status > System Details** dialog:



System Details

System Name	awplus
System Contact	
System Location	

Description

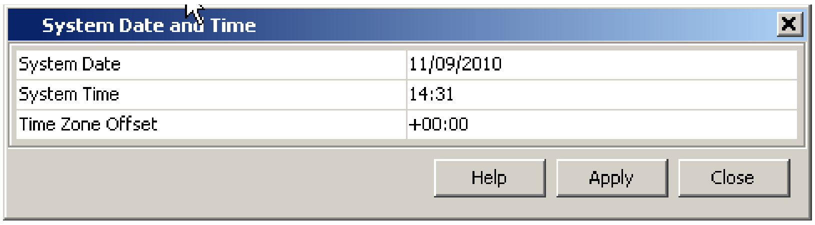
Label / Field / Button	Description
`System Name`	Enter the network name of the system.
`System Contact`	Enter the contact information for the system, from 0 to 255 characters long. Valid characters are any printable characters and spaces.
`System Location`	Enter the location of the system, from 0 to 255 characters long. Valid characters are any printable characters and spaces.

System > Status > System Date and Time

The **System > Status > System Date and Time** dialog allows you to configure the date and time for the switch.

Configuration Dialog

Figure C-3: Example showing **System > Status > System Date and Time** dialog:



System Date and Time

System Date	11/09/2010
System Time	14:31
Time Zone Offset	+00:00

Description

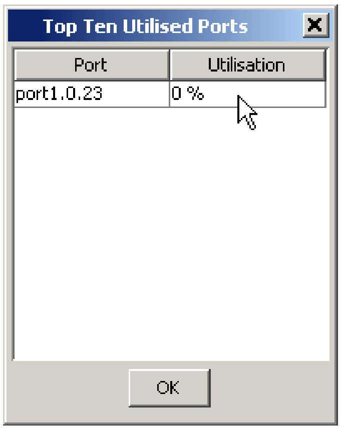
Label / Field / Button	Description
`System Date`	Enter the current system date in month, day, and year format.
`System Time`	Enter the local time for the system clock in hours and minutes.
`Time Zone Offset`	Enter the offset to the UTC (Coordinated Universal Timezone) for a local timezone in hours and minutes.

System > Status > Top Ten Utilised Ports

The **System** > **Status** > **Top Ten Utilised Port** dialog allows you to monitor port utilisation on the switch.

Configuration Dialog

Figure C-4: Example showing **System > Status > Top Ten Utilised Ports** dialog:



Port	Utilisation
port1.0.23	0 %

Description

Label / Field / Button	Description
`Port`	Displays up to ten ports that are used the most on the switch. You can sort by ascending or descending port order.
`Utilisation`	Displays the utilisation percentage for the port. You can sort by ascending or descending utilisation percentage.

System > Identity

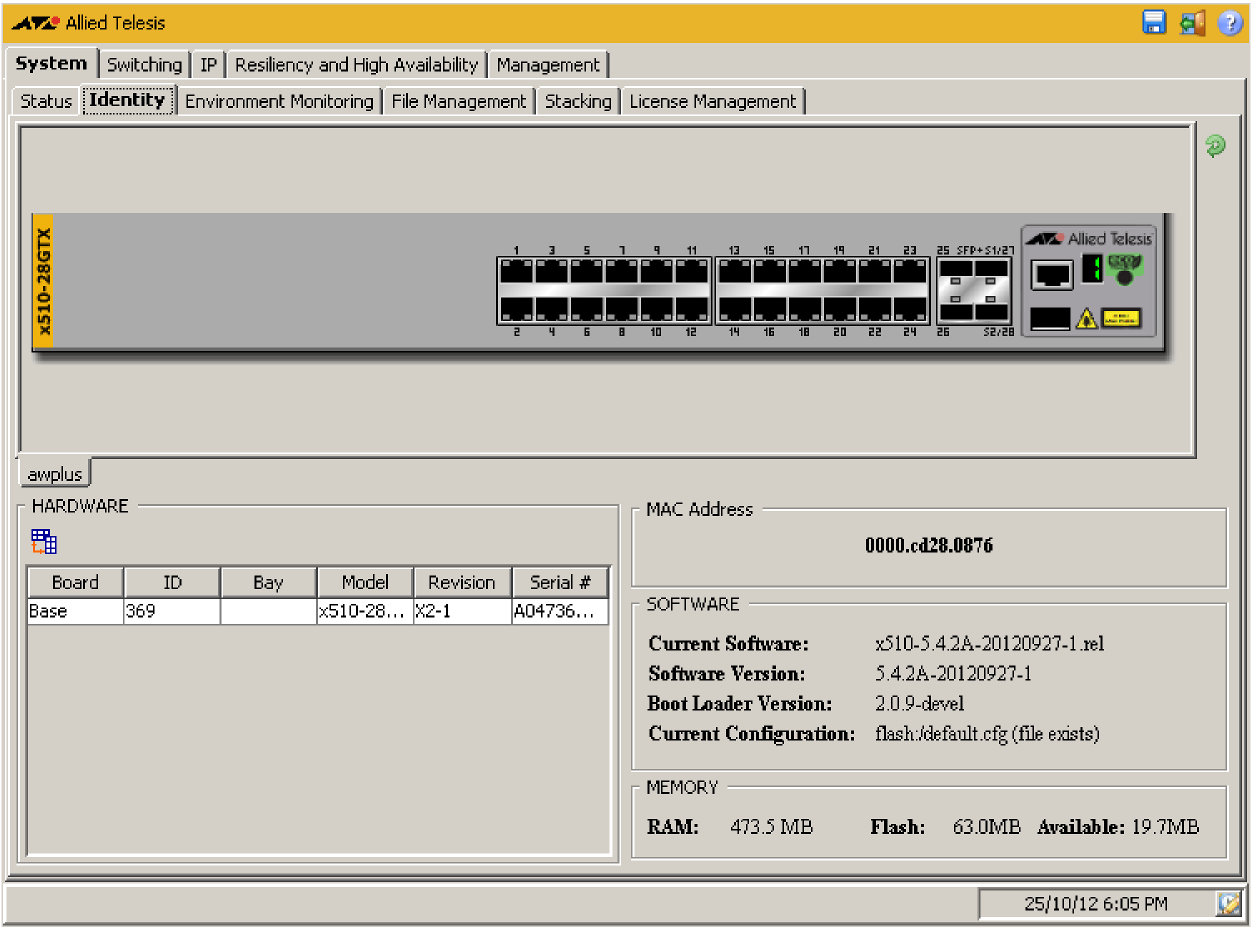
The **System > Identity** menu tab displays physical properties, software version and configuration file name.

Note For systems equipped and configured using VCStack there is a separate tab for each stack member with the system name displayed on each tab.

The last two SFP+ port LEDs are lit when VCStack is enabled on the switch.

Menu Tab

Figure C-5: Example showing the **System > Identity** menu tab:



HARDWARE

Board	ID	Bay	Model	Revision	Serial #
Base	369		x510-28...	X2-1	A04736...

MAC Address

0000.cd28.0876

SOFTWARE

Current Software: x510-5.4.2A-20120927-1.rel
Software Version: 5.4.2A-20120927-1
Boot Loader Version: 2.0.9-devel
Current Configuration: flash:/default.cfg (file exists)

MEMORY

RAM: 473.5 MB **Flash:** 63.0MB **Available:** 19.7MB

25/10/12 6:05 PM

Description

Label / Field / Button	Description
`HARDWARE`	Displays the board, ID, bay, model, revision and serial number of the switch main board and any installed XEMs.
`MAC Address`	Displays the MAC Address of the switch in hexadecimal in the format `HHHH.HHHH.HHHH`.

Label / Field / Button(cont.)	**Description**(cont.)
`SOFTWARE`	Displays the software release file name, software version, boot loader version, and configuration file name loaded on the switch.
`MEMORY`	Displays the amount of installed RAM and Flash, plus the remaining Flash available on the switch.

System > Environment Monitoring

The **System > Environment Monitoring** menu tab allows you to display the status of the environmental properties, such as all voltages and temperatures, which the system monitors.

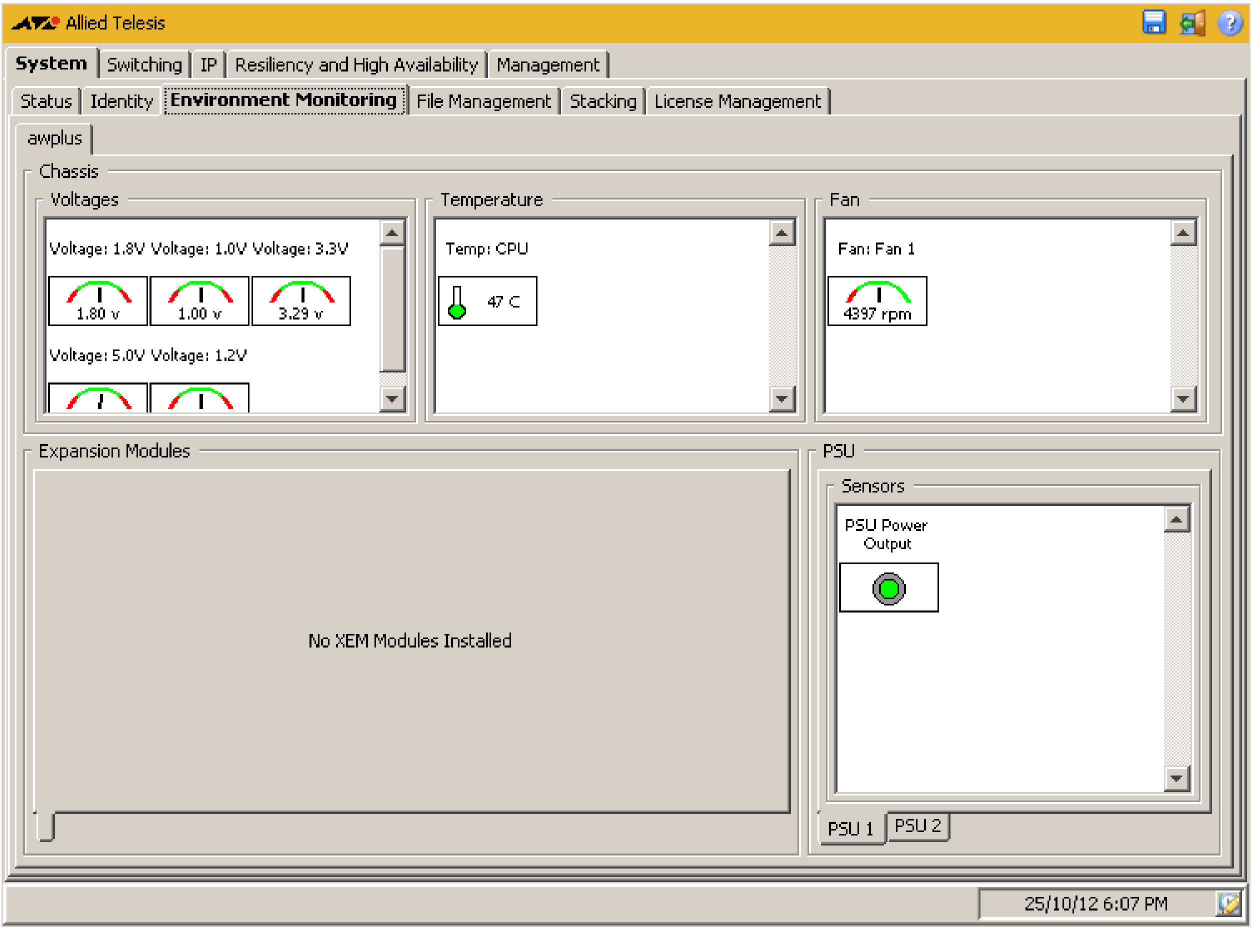
Note For systems equipped and configured using VCStack there is a separate tab for each stack member with the system name displayed on each tab.

The last two SFP+ port LEDs are lit when VCStack is enabled on the switch.

Figure C-6:

Menu Tab

Figure C-7: Example showing the **System > Environment Monitoring** menu tab:



Voltage: 1.8V Voltage: 1.0V Voltage: 3.3V
1.80 v 1.00 v 3.29 v
Voltage: 5.0V Voltage: 1.2V

Temp: CPU
47 C

Fan: Fan 1
4397 rpm

No XEM Modules Installed

PSU Power Output

25/10/12 6:07 PM

Description

Label / Field / Button	Description
`Chassis`	Displays the operational status of chassis voltages and temperatures for the switch.
`Fan`	Displays the operational status of the switch fans.

Label / Field / Button(cont.)	**Description**(cont.)
`XEM`	Displays the operational status of voltages, temperatures, and fans for any installed XEMs.
`PSU`	Displays the operational status of temperatures and fans for any installed pluggable PSUs.

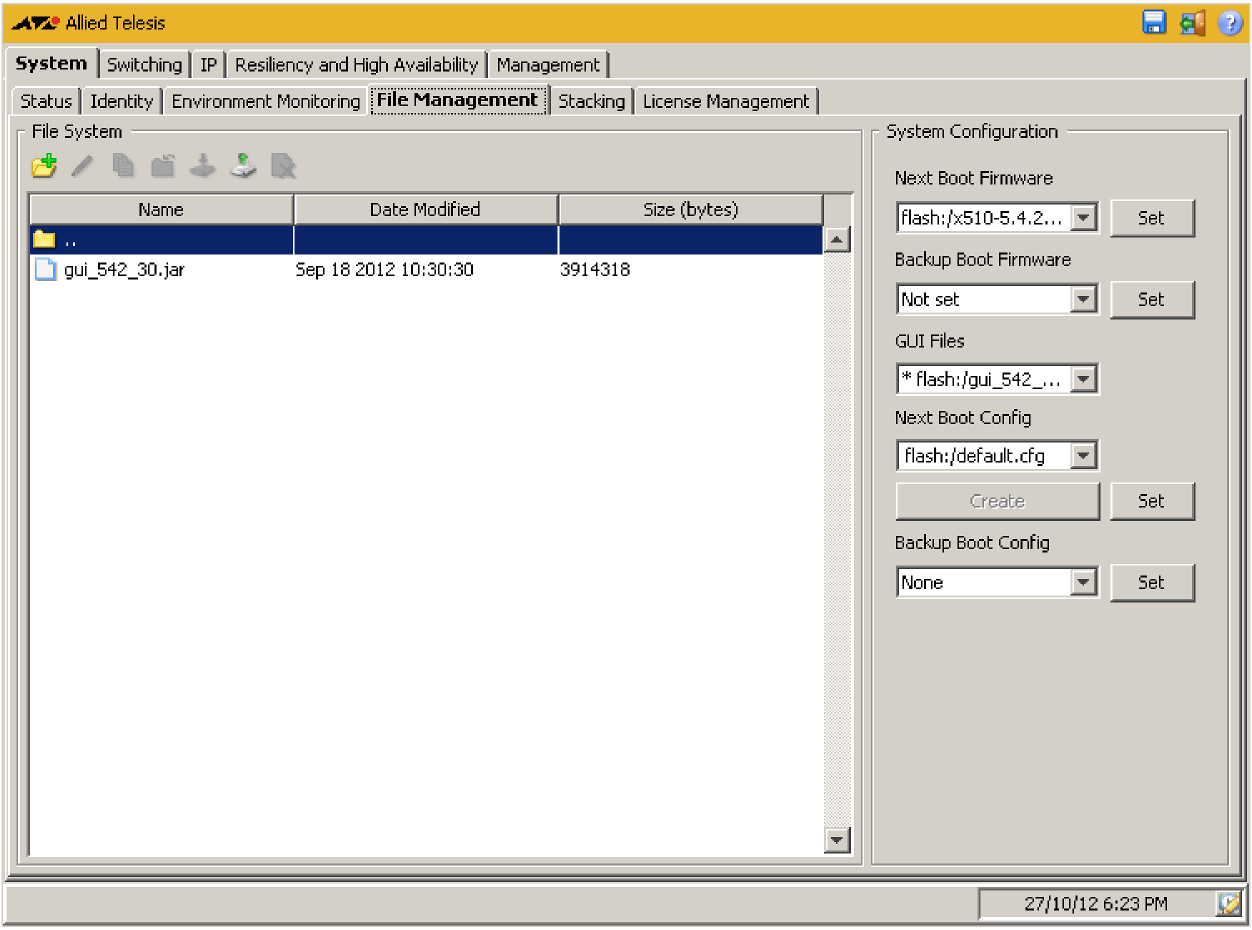
System > File Management

The **System > File Management** menu tab allows you to create, copy, delete, upload or download boot and backup release and configuration files to and from the switch.

You can specify fallback or backup release and configuration files in case the boot release or configuration files become corrupted, and you can also specify the boot release and configuration files to boot directly from a USB flash drive or to boot from flash.

Menu Tab

Figure C-8: Example showing the **System > File Management** menu tab:



Name	Date Modified	Size (bytes)
..		
gui_542_30.jar	Sep 18 2012 10:30:30	3914318

Next Boot Firmware: flash:/x510-5.4.2...

Backup Boot Firmware: Not set

GUI Files: * flash:/gui_542_...

Next Boot Config: flash:/default.cfg

Backup Boot Config: None

Description: File System

Label / Field / Button	Description
`File System`	Displays file names, file dates, and file sizes of files in Flash, NVS or USB flash drive. Note that only USB flash drive files that are resident on the stack master are shown. Also, only files that have a total URL length of 112 characters or less are displayed. The URL is the path to the file and is of the form `<hostname>-<stack_id>/<filesystem>:/<pathname>`, for example, `awplus-1/flash:/test.cfg`. The GUI will immediately show all file changes to the NVS and Flash filesystems, regardless of how they have been made, either via the GUI or CLI. However, the USB flash drive filesystem is treated differently as it is not permanently mounted. The GUI will only update USB flash drive files when this is inserted or deleted, or when the changes are made via the GUI. They are not updated if modified via the CLI. The buttons shown below the File System label also allow you upload, download, move, copy, and delete files respectively.
`File System / Add Folder`	Select the folder you want to create a new sub-folder in then click on the Add Folder button located directly below the File System label.
`File System / Rename File`	Select the file you want to rename then click on the Rename File or Folder button located directly below the File System label.
`File System / Copy File`	Select the file you want to copy then click on the Copy File button located directly below the File System label. Choose the Destination Folder from the drop down list in the Copy File dialog then select OK to copy the file to the chosen destination.
`File System / Move File`	Select the file you want to move then click on the Move File button located directly below the File System label. Choose the Destination Folder from the drop down list in the Move File dialog then select OK to move the file to the chosen destination.
`File System / Download File`	Select the file you want to download then click on the Download File button located directly below the File System label.
`File System / Upload File`	Click on the Upload File button located directly below the File System label then select the file you want to upload.
`File System / Delete File`	Select the file or folder you want to delete then click on the Delete File button located directly below the File System label.

Description: System Configuration

Label / Field / Button	Description
`System Configuration`	Configures running and backup software, GUI software, and configuration files in Flash or card memory available on the switch.
`System Configuration / Next Boot Firmware`	Choose the Next Boot Firmware `.rel` file and path from the drop down list then click Set to make this file the firmware that starts after reboot. You can set a Next Boot Firmware `.rel` file to boot directly from a USB flash drive.

Label / Field / Button(cont.)	**Description**(cont.)
`System Configuration / Backup Boot Firmware`	Choose the Backup Boot Firmware `.rel` file and path from the drop down list then click Set to boot from this file at reboot. A Backup Boot Firmware .rel file is used instead of the Next Boot Firmware `.rel` file if the Next Boot Firmware `.rel` file is corrupted.
`System Configuration / GUI Files`	Displays the GUI file name and file location on the switch and indicates the currently running GUI file with a prefixed asterisk (e.g. `* flash:/gui_542_33.jar`). Note that you cannot set the GUI version from within the GUI itself. See the GUI installation instructions in *Appendix C: GUI Reference* of the current *AlliedWare Plus Software Reference* to install GUI files. The latest version of the GUI .jar file loaded is run by the switch automatically.
`System Configuration / Next Boot Config`	Choose the Next Boot Config `.cfg` file and path from the drop down list then click Set to make this file the Config `.cfg` file that the switch uses at reboot. From 5.4.1 release onwards, you can set a Next Boot Config `.cfg` file to load directly from a USB flash drive.
`System Configuration / Backup Boot Config`	Choose the Backup Boot Config Files .cfg file and path from the drop down list then click Set to make boot from this file at reboot. A Backup Boot Config `.cfg` file is used instead of the Next Boot config `.cfg` file if the Next Boot Config `.cfg` file is corrupted.

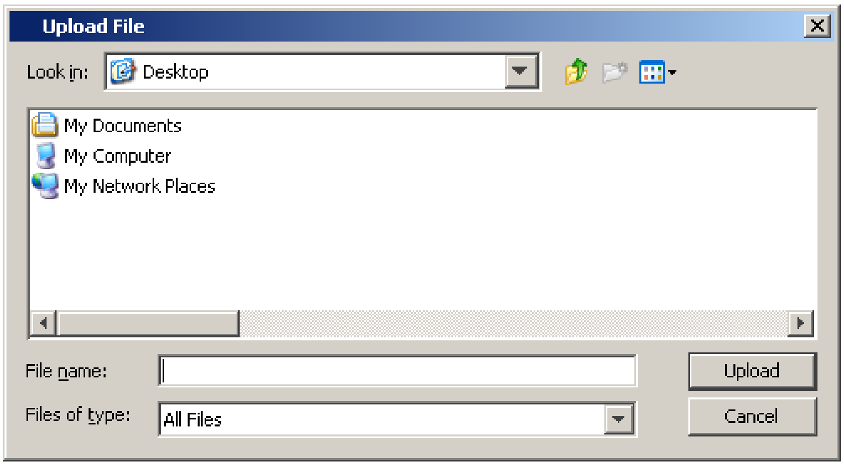
System > File Management > Upload File

The **System** > **File Management** > **Upload File** dialog allows you to upload files (e.g. release and configuration files) from a client device to the switch. Select the Upload File button below the File System label to access this dialog.

Note The Upload file option, select from the row of small icons, the icon that is located 5th from the left. If the icon row is greyed out, selecting a file from the window should return their display.

Configuration Dialog

Figure C-9: Example showing the **System > File Management > Upload File** dialog



Description

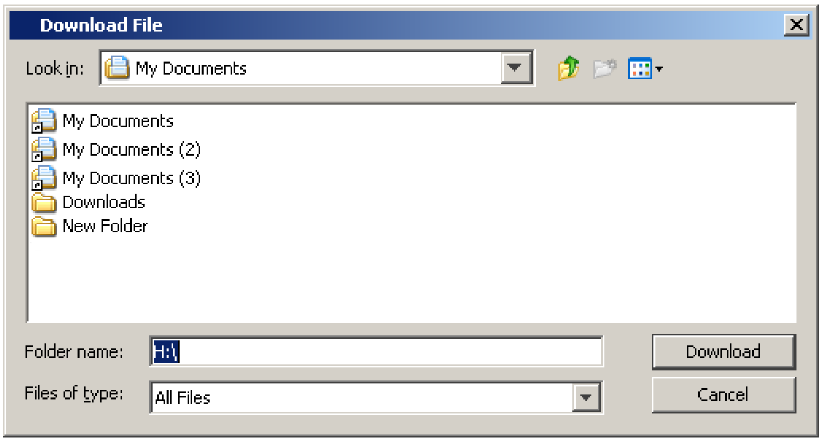
Label / Field / Button	Description
`Destination Folder`	Select the destination folder to upload the selected file from. An uploading progress bar displays after clicking the Upload button.

System > File Management > Download File

The **System** > **File Management** > **Download File** dialog allows you to download files (e.g. release and configuration files) from the switch to a client device. Select the Download File button below the File System label to access this dialog.

Configuration Dialog

Figure C-10: Example showing the **System > File Management > Download File** dialog



Look in: My Documents

Folder name: H:\

Files of type: All Files

Description

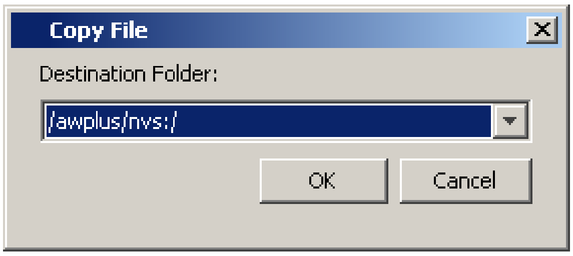
Label / Field / Button	Description
`Destination Folder`	Select the destination folder to download the selected file to. A downloading progress bar displays after clicking the Download button.

System > File Management > Copy File

The **System** > **File Management** > **Copy File** dialog allows you to copy files (e.g. release and configuration files). Select the Copy File button below the File System label to access this dialog.

Configuration Dialog

Figure C-11: Example showing the **System > File Management > Copy File** dialog



Destination Folder: /awplus/nvs:/

Description

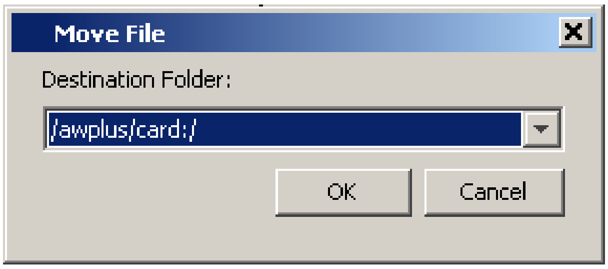
Label / Field / Button	Description
`Destination Folder`	Select the destination folder to copy the selected file to. A copying progress bar displays after clicking the OK button.

System > File Management > Move File

The **System** > **File Management** > **Move File** dialog allows you to move files (e.g. release and configuration files). Select the Move File button below the File System label to access this dialog.

Configuration Dialog

Figure C-12: Example showing the **System > File Management > Move File** dialog



Description

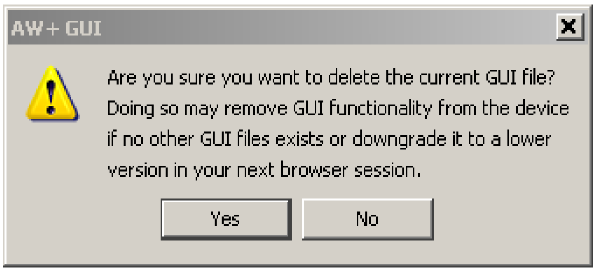
Label / Field / Button	Description
`Destination Folder`	Select the destination folder to move the selected file to. A moving progress bar displays after clicking the OK button.

System > File Management > Delete File

The **System** > **File Management** > **Delete File** dialog allows you to delete files (e.g. release and configuration files). Select the Delete File button below the File System label to access this dialog.

Configuration Dialog

Figure C-13: Example showing **System > File Management > Delete File** dialog:



Are you sure you want to delete the current GUI file? Doing so may remove GUI functionality from the device if no other GUI files exists or downgrade it to a lower version in your next browser session.

Description

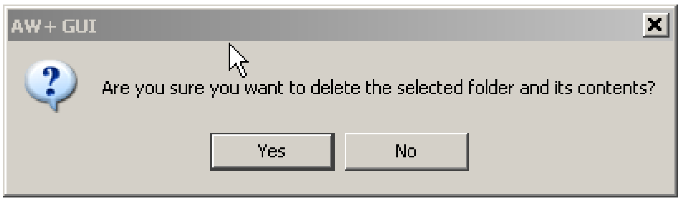
Label / Field / Button	Description
`Yes`	Confirm selected file deletion operation.
`No`	Cancel selected file deletion operation.

System > File Management > Delete Folder

The **System** > **File Management** > **Delete Folder** dialog allows you to delete folders in the flash or USB flash drive file system containing files (e.g. release and configuration files). Select the Delete Folder button below the File System label to access this dialog.

Configuration Dialog

Figure C-14: Example showing **System > File Management > Delete Folder** dialog:



Are you sure you want to delete the selected folder and its contents?

Description

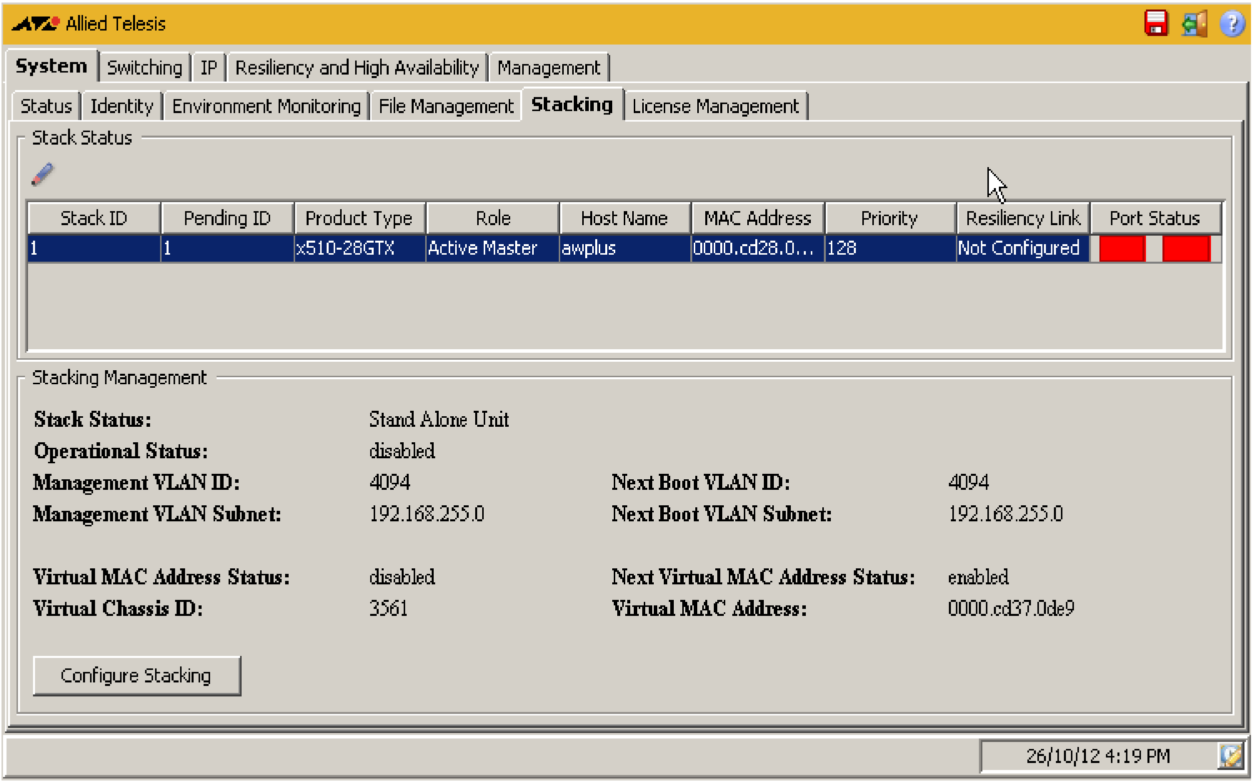
Label / Field / Button	Description
`Yes`	Confirm selected folder deletion operation.
`No`	Cancel selected folder deletion operation.

System > Stacking

The **System > Stacking** menu tab allows you to display and monitor a summary of the identity and status of stack members, plus you can also configure the VLAN ID and IP subnets used for internal VCStack communication.

Menu Tab

Figure C-15: Example showing the **System > Stacking** menu tab:



Stack Status

Stack ID	Pending ID	Product Type	Role	Host Name	MAC Address	Priority	Resiliency Link	Port Status
1	1	x510-28GTX	Active Master	awplus	0000.cd28.0...	128	Not Configured	🟥 🟥

Stacking Management

Stack Status: Stand Alone Unit
Operational Status: disabled
Management VLAN ID: 4094 **Next Boot VLAN ID:** 4094
Management VLAN Subnet: 192.168.255.0 **Next Boot VLAN Subnet:** 192.168.255.0

Virtual MAC Address Status: disabled **Next Virtual MAC Address Status:** enabled
Virtual Chassis ID: 3561 **Virtual MAC Address:** 0000.cd37.0de9

Configure Stacking

26/10/12 4:19 PM

Description: Stacking Management

Label / Field / Button	Description
`Stacking Management / Stack Status`	The stack's overall status. Note that a warning is issued if the stack is not connected in a standard ring topology.
`Stacking Management / Operational Status`	The operational status of the stack. Can be: enabled (1), disabled (2).
`Stacking Management / Management VLAN ID`	The VLAN ID currently used for stack management. The default stack management VLAN ID is 4094.
`Stacking Management / Next Boot VLAN ID`	The VCS management VLAN ID to be assigned after the next reboot.
`Stacking Management / Management VLAN Subnet`	The VLAN subnet currently used for stack management.
`Stacking Management / Next Boot VLAN Subnet`	The stacking management VLAN subnet address after rebooting.

Label / Field / Button(cont.)	**Description**(cont.)
`Stacking Management / Virtual MAC Address Status`	Indicates whether the virtual MAC address is enabled or disabled. Note that when running VCStack configurations you must *enable* Virtual MAC addressing.
`Stacking Management / Next Virtual MAC Address Status`	Indicates whether the next virtual MAC address is enabled or disabled. Note that when running VCStack configurations you must *enable* Virtual MAC addressing.
`Stacking Management / Virtual Chassis ID`	Displays the current virtual chassis ID.
`Stacking Management / Virtual MAC Address`	Displays the virtual MAC address used by the stack.
`Configure Stacking`	Configures the VCS management VLAN ID, the subnet address of the VCS management VLAN, and the Virtual MAC Address Status.

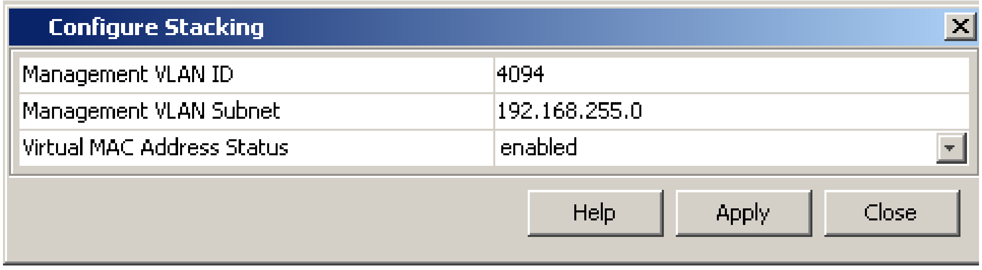
Description: Stack Status

Label / Field / Button	Description
`Stack Status / Stack ID`	The Stack member ID.
`Stack Status / Pending ID`	The Stack member ID to be assigned to the device after the next reboot.
`Stack Status / Product Type`	The Stack member product type; for example, SwitchBlade x908.
`Stack Status / Role`	Stack member's role in the stack (either Active Master or Backup Member).
`Stack Status / Host Name`	The host name of the Stack member.
`Stack Status / MAC Address`	Stack member's hardware MAC address. Note that frames from devices within a stacked virtual chassis will carry the source address of the stack master.
`Stack Status / Priority`	The priority for election of stack master (0 to 255). The lowest number has the highest priority. Note that where stack members have the same priority setting, the switch with the lowest MAC address will become the stack master.
`Stack Status / Resiliency Link`	Status of the stack members resiliency link. Can be one of: `Configured`, `Successful`, `Failed`, `Not Configured`.
`Stack Status / Port Status`	When the rectangle is colored GREEN, it means that the port has a learned neighbor connected. Note that the number in the rectangle indicates the stack ID of the learned neighbor connected to the port. When the rectangle is colored BLACK then the port status is down. When the rectangle is colored RED then Operational Status for the port has been set to disabled.

System > Stacking > Configure Stacking

The **System > Stacking > Configure Stacking** dialog allows you to configure the VLAN ID and IP subnets used for internal VCStack communication.

Figure C-16:



Configure Stacking	
Management VLAN ID	4094
Management VLAN Subnet	192.168.255.0
Virtual MAC Address Status	enabled

Description

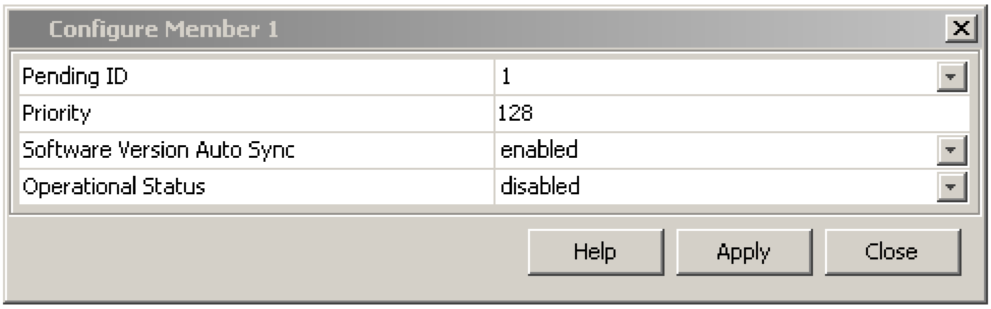
Label / Field / Button	Description
`Management VLAN ID`	Enter the VLAN ID for stack management. The default stack management VLAN ID is 4094.
`Management VLAN subnet`	Enter the subnet address of the VCS management VLAN.
`Virtual MAC Address Status`	Select the status of the Virtual MAC Addressing feature, which can be set to `disabled` or `enabled` from the drop down list. Note that when running VCStack configurations, you must enable Virtual MAC Addressing.

System > Stacking > Configure Stack Member

The **System > Stacking > Configure Stack Member** dialog allows you to configure the Pending ID, Priority, Software Version Auto Synchronization and Operational Status used for internal VCStack communication. To display this dialog box select the appropriate stack member icon from the **Stack Status** panel.

Configuration Dialog

Figure C-17: Example showing **System > Stacking > Configure Stack Member** :



Configure Member 1	
Pending ID	1
Priority	128
Software Version Auto Sync	enabled
Operational Status	disabled

Description

Label / Field / Button	Description
`Pending ID`	Enter the Pending ID for the stack member.
`Priority`	Enter the Priority for the stack member.
`Software Version Auto Sync`	Select the enabled or disabled options to enable or disable the Software Version Auto Synchronization feature for the stack member.
`Operational Status`	Select the enabled or disabled options to enable or disable the stack member.

System > License Management

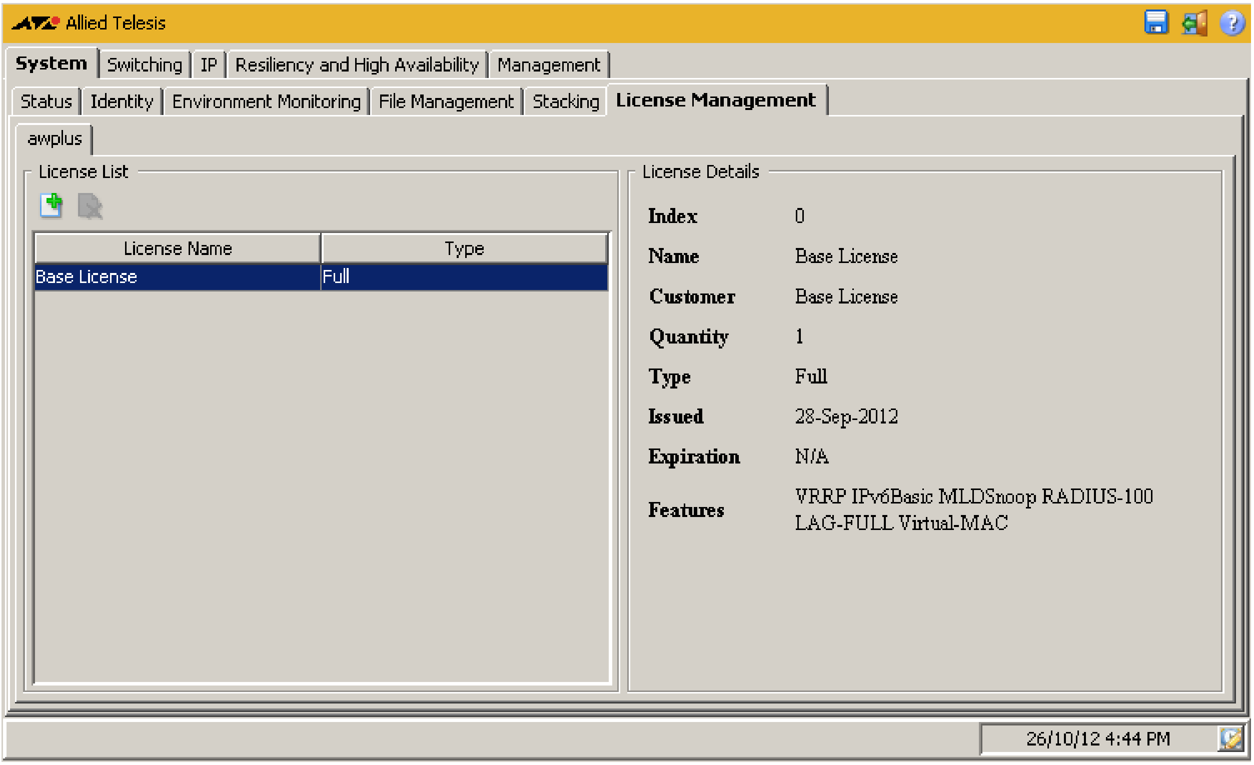
The **System > License Management** menu tab allows you to view, add and delete feature licenses.

Note For systems equipped and configured using VCStack there is a separate tab for each stack member with the system name displayed on each tab.

The last two SFP+ port LEDs are lit when VCStack is enabled on the switch.

Note If a license is added to, or deleted from, a stack member then the same action must be taken on all other stack members. Otherwise incompatible licensing will occur and affected devices will not rejoin the stack following a reboot.

Menu Tab Figure C-18: Example showing the **System > License Management** tab:



License Name	Type
Base License	Full

Index	0
Name	Base License
Customer	Base License
Quantity	1
Type	Full
Issued	28-Sep-2012
Expiration	N/A
Features	VRRP IPv6Basic MLDSnoop RADIUS-100 LAG-FULL Virtual-MAC

Description

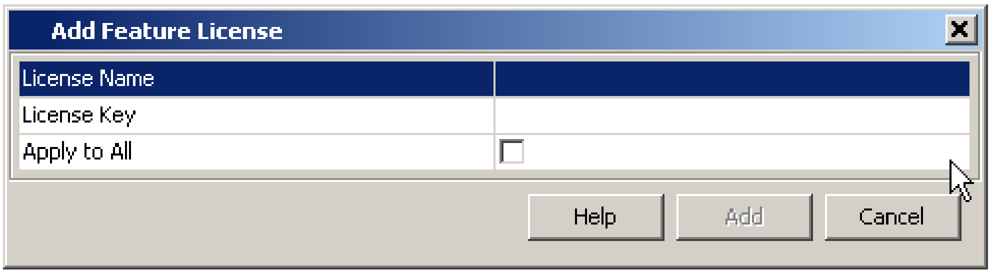
Label / Field / Button	Description
`License List / License Name`	Name of the license bundle.
`License List / Type`	The type of license activated on the switch: full or temporary.
`License Details / Index`	Index identifying entry.
`License Details / Name`	Name of the license bundle.
`License Details / Customer`	Customer name.
`License Details / Quantity`	Quantity of licenses included in the feature key.
`License Details / Type`	Full or temporary license types.
`License Details / Issued`	Date the key was generated.
`License Details / Expiration`	Expiry date for a temporary license.
`License Details / Features`	List of features enabled by the license.

System > License Management > Add Feature License

The **System > License Management > Add Feature License** dialog allows you add feature licenses by specifying the license name and the license key. You can add a license for all the stack members or for a single stack member.

Configuration Dialog

Figure C-19: Example showing **System > License Management > Add Feature License** :



Add Feature License

License Name	
License Key	
Apply to All	□

Description

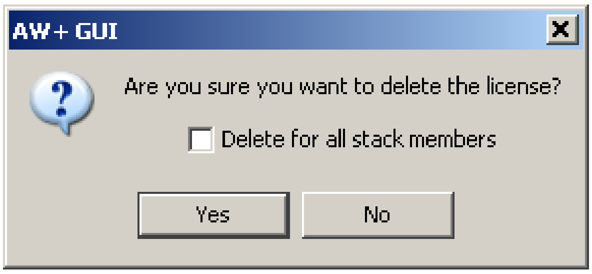
Label / Field / Button	Description
`License Name`	Enter the license name of the software feature.
`License Key`	Enter the encrypted license key to enable this software feature.
`Apply to All`	Select the checkbox to apply the license to all the stack members.

System > License Management > Delete Feature License

The **System** > **License Management** > **Delete Feature License** dialog allows you delete feature licenses by specifying the license name and the license key. You can delete a license for all the stack members or for a single stack member.

Configuration Dialog

Figure C-20: Example showing **System > License Management > Delete Feature License** :



AW+ GUI

Are you sure you want to delete the license?

□ Delete for all stack members

Description

Label / Field / Button	**Description**
`Delete for all stack members`	Select the checkbox to delete the license for all the stack members.

Switching > Ports

The **Switching > Ports** menu tab allows you to view, and configure Layer 1 properties:

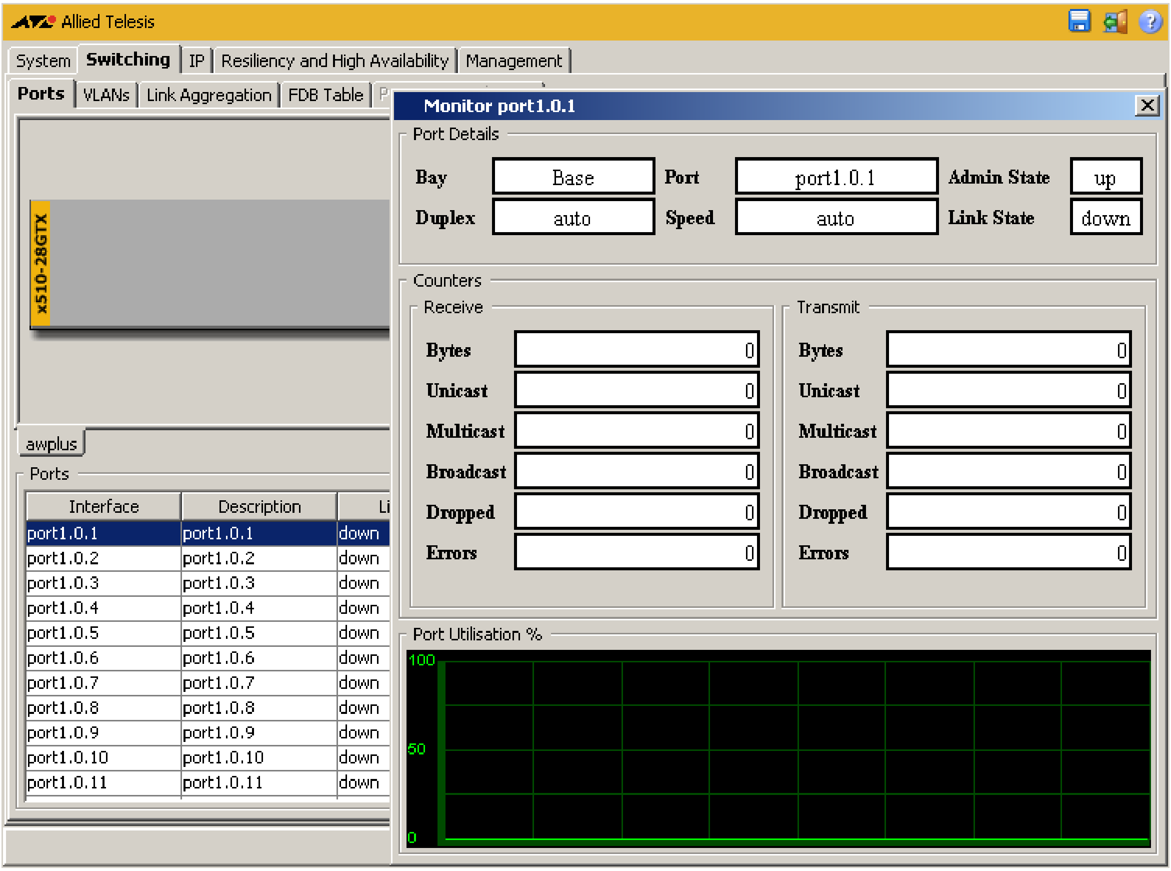
- Right-clicking a port allows you to select monitoring or configuration dialogs for the selected port.
- The monitoring dialog displays port status, statistics and a brief utilization history.
- The configuration dialog allows you to configure Administrative State, Auto Negotiation, Speed and Duplex settings for the port.

Note Speed and Duplex settings can only be changed if Auto Negotiation is disabled.

Note For systems equipped and configured using VCStack there is a separate tab for each stack member with the system name displayed on each tab.

The last two SFP+ port LEDs are lit when VCStack is enabled on the switch.

Menu Tab Figure C-21: Example showing the **Switching > Ports** menu tab



Port Details

Bay	Base	Port	port1.0.1	Admin State	up
Duplex	auto	Speed	auto	Link State	down

Counters

Receive		Transmit	
Bytes	0	Bytes	0
Unicast	0	Unicast	0
Multicast	0	Multicast	0
Broadcast	0	Broadcast	0
Dropped	0	Dropped	0
Errors	0	Errors	0

Ports

Interface	Description	Li
port1.0.1	port1.0.1	down
port1.0.2	port1.0.2	down
port1.0.3	port1.0.3	down
port1.0.4	port1.0.4	down
port1.0.5	port1.0.5	down
port1.0.6	port1.0.6	down
port1.0.7	port1.0.7	down
port1.0.8	port1.0.8	down
port1.0.9	port1.0.9	down
port1.0.10	port1.0.10	down
port1.0.11	port1.0.11	down

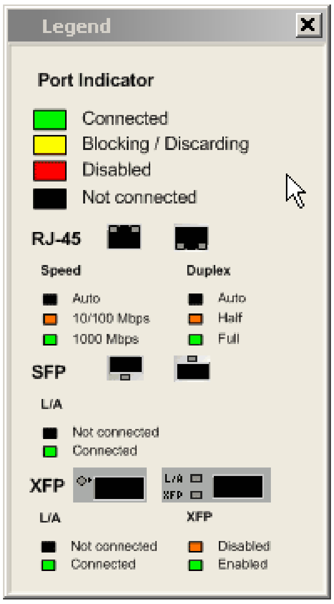
Description

Label / Field / Button	Description
`Ports`	Displays port number, description of the port, link and administrative status, duplex mode, speed and uptime (in milliseconds) for the selected port.

Legend

Figure C-22: Example showing **Switching > Ports > Legend**:

To select the Legend panel click the View Legend icon (white L within a blue circle) on the top right of the Switching - Ports screen.

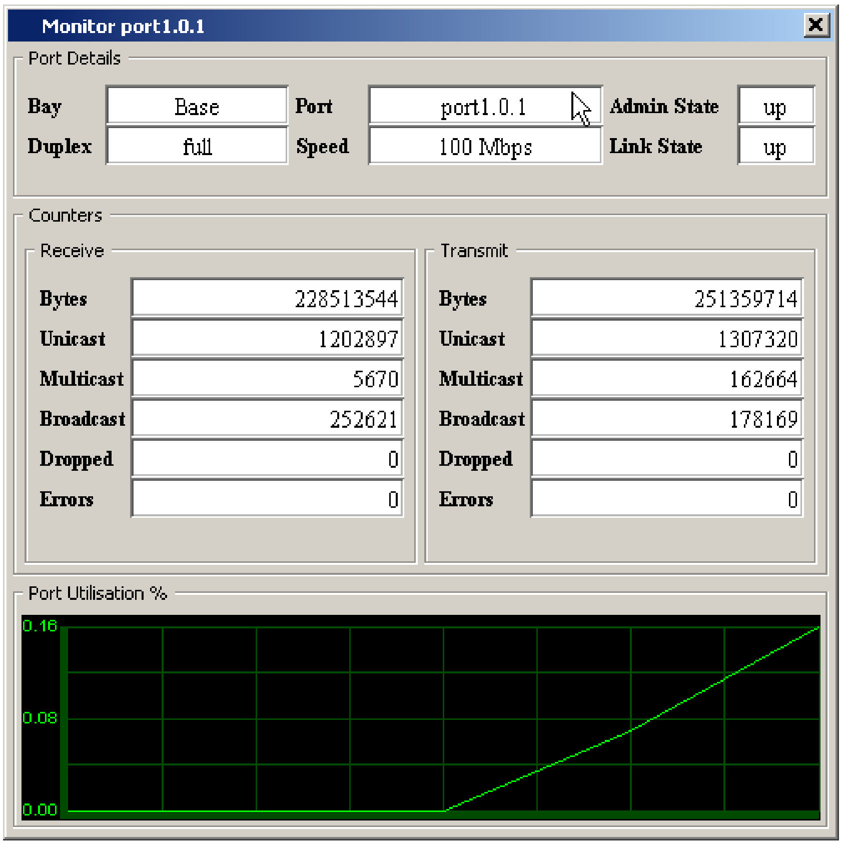


Switching > Ports > Monitor Port

The **Switching > Ports > Monitor Port** dialog allows you monitor port counters.

Configuration Dialog

Figure C-23: Example showing the **Switching > Ports > Monitor Port** dialog:



Monitor port1.0.1

Port Details

Bay	Base	Port	port1.0.1	Admin State	up
Duplex	full	Speed	100 Mbps	Link State	up

Counters

Receive		Transmit	
Bytes	228513544	Bytes	251359714
Unicast	1202897	Unicast	1307320
Multicast	5670	Multicast	162664
Broadcast	252621	Broadcast	178169
Dropped	0	Dropped	0
Errors	0	Errors	0

Port Utilisation %

Description

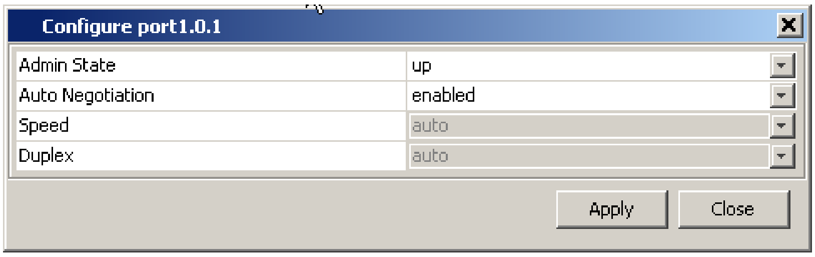
Label / Field / Button	Description
`Port Details`	Monitors the bay, port, duplex, speed, administrative state and link states for the selected port.
`Counters`	Monitors the counters for bytes received/transmitted, unicast packets received/transmitted, multicast packets received/transmitted, broadcast packets received/transmitted, dropped packets received/transmitted, and errors received/transmitted for the selected port.
`Port Utilisation%`	Monitors and graphs the usage percentage for the selected port.

Switching > Ports > Configure Port

The **Switching > Ports > Configure Port** dialog allows you configure Administrative State, Auto Negotiation, Speed and Duplex settings for the selected port.

Configuration Dialog

Figure C-24: Example showing the **Switching > Ports > Configure Port** dialog:


Configure port1.0.1

Admin State	up
Auto Negotiation	enabled
Speed	auto
Duplex	auto

Description

Label / Field / Button	Description
`Admin State`	Select `up` or `down` from the drop down list in this dialog to specify the administrative state for the selected port.
`Auto Negotiation`	Select `disabled` or `enabled` from the drop down list in this dialog to specify auto negotiation for the selected port. Note that selecting `enabled` to enable `Auto Negotiation` will disable Speed and Duplex options, indicated by greyed out options.
`Speed`	Select `10 Mbps`, `100 Mbps`, `1000 Mbps`, `10 Gbps`, or `auto` from the drop down list in this dialog to specify the speed setting for the selected port. Note that the options for speed settings are only available if `Auto Negotiation` has been `disabled` for the selected port.
`Duplex`	Select `full`, `half`, or `auto` from the drop down list in this dialog to specify the duplex setting for the selected port. Note that the options for duplex settings are only available if `Auto Negotiation` has been `disabled` for the selected port.

Switching > VLANs

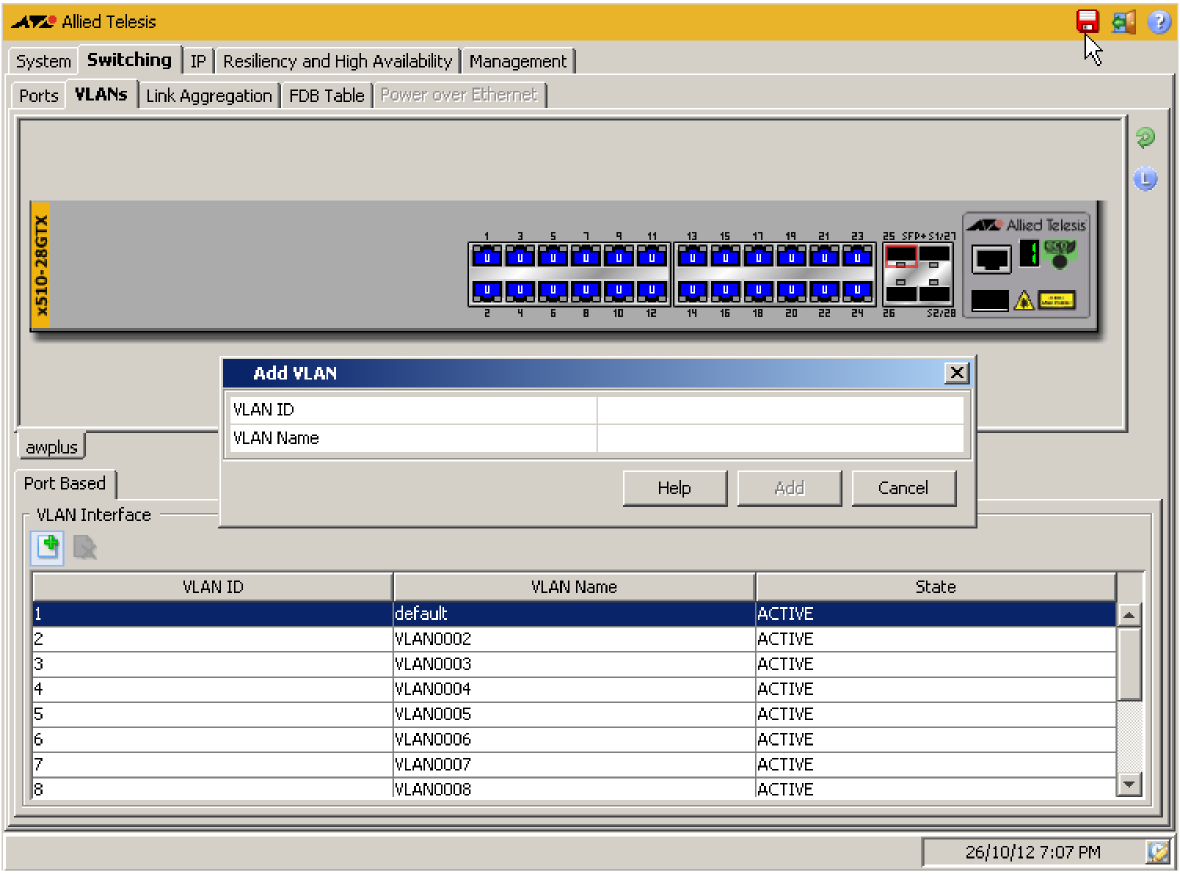
The **Switching** > **VLANs** menu tab allows you to view, and configure Layer 2 properties:

- Right-clicking a port allows you to select a VLAN to be tagged or untagged for the port, or to remove a port from the VLAN.
- Define VLANs before assigning VLANs to ports on the front panel of the switch.
- Selecting the **+** icon (under the VLAN Interface label on the VLAN tab below the front panel illustration) allows you to add a VLAN by specifying the VLAN ID and VLAN Name.
- Selecting the **x** icon (under the VLAN Interface label below the front panel illustration) allows you to delete a VLAN (except for the default VLAN 1 that is assigned to all ports).

Note For systems equipped and configured using VCStack there is a separate tab for each stack member with the system name displayed on each tab.

The last two SFP+ port LEDs are lit when VCStack is enabled on the switch.

Menu Tab Figure C-25: Example showing the **Switching > VLANs** menu tab:



Add VLAN

VLAN ID	
VLAN Name	

VLAN ID	VLAN Name	State
1	default	ACTIVE
2	VLAN0002	ACTIVE
3	VLAN0003	ACTIVE
4	VLAN0004	ACTIVE
5	VLAN0005	ACTIVE
6	VLAN0006	ACTIVE
7	VLAN0007	ACTIVE
8	VLAN0008	ACTIVE

26/10/12 7:07 PM

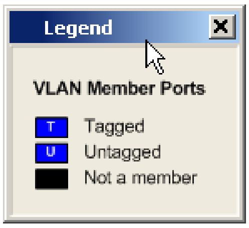
Description

Label / Field / Button	Description
`Port Based / VLAN ID`	The VID of the VLAN that is enabled or disabled in the range 1-4094.
`Port Based / VLAN Name`	The ASCII name of the VLAN with a maximum length of 32 characters.
`Port Based / State`	The state of the VLAN, either enabled ('**ACTIVE**' displayed) or disabled ('**INACTIVE**' displayed).

Legend

Figure C-26: Example showing **Switching > VLANs > Legend**:

To select the Legend panel click the View Legend icon (white L within a blue circle) on the top right of the Switching - Ports screen.



Legend ☒

VLAN Member Ports

T Tagged

U Untagged

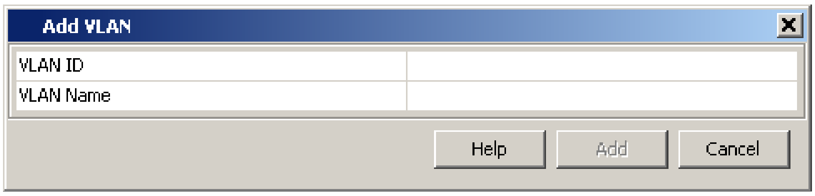
■ Not a member

Switching > VLANs > Add VLAN

The **Switching > VLANs > Add VLAN** dialog allows you add a VLAN by specifying the VLAN ID and VLAN Name.

Configuration Dialog

Figure C-27: Example showing the **Switching > VLANs > Add VLAN** dialog:



Add VLAN

VLAN ID	
VLAN Name	

Description

Label / Field / Button	Description
`VLAN ID`	Enter the VID of the VLAN that is enabled or disabled in the range <1-4094>.
`VLAN Name`	Enter the ASCII name of the VLAN with a maximum length of 32 characters.

Switching > Link Aggregation

The **Switching > Link Aggregation** menu tab allows you to view, and configure Layer 2 properties:

- Right-clicking a port allows you to select assign or remove the port to a Static Channel or a Dynamic Channel (LACP - Link Aggregation Control Protocol) group.
- Define Static Channel or Dynamic Channel (LACP) groups before assigning them to ports on the front panel of the switch.
- Selecting the + icon (located below the front panel illustration of your switch) allows you to add a Static Channel or Dynamic Channel (LACP) group by specifying the Channel ID.

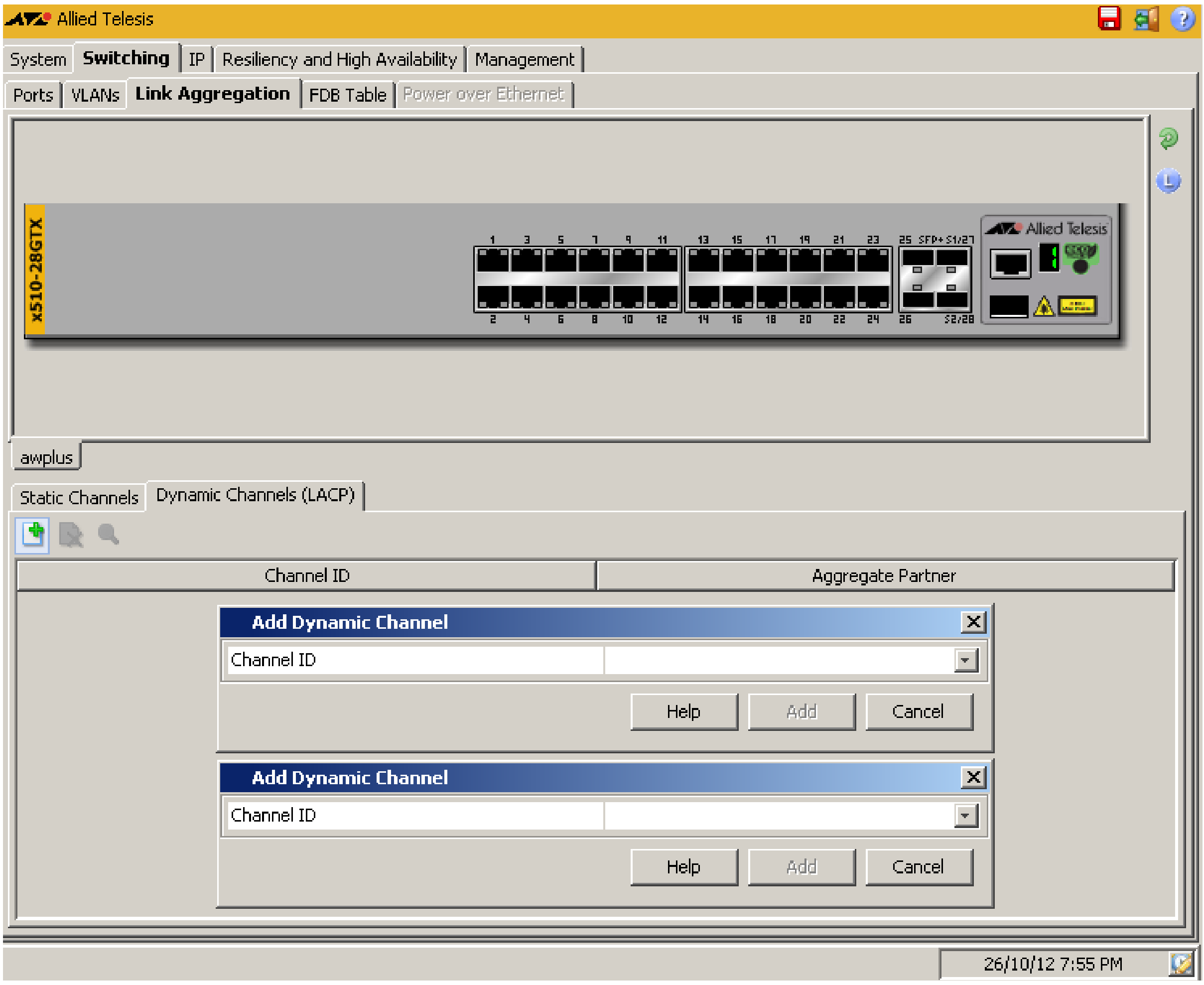
Note Up to 96 Static Channel groups and up to 32 Dynamic Channel (LACP) groups can be defined on a switch.

- Selecting the x icon (located below the front panel illustration of your switch) allows you to delete a Static Channel or Dynamic Channel (LACP) group.

Note For systems equipped and configured using VCStack there is a separate tab for each stack member with the system name displayed on each tab.

The last two SFP+ port LEDs are lit when VCStack is enabled on the switch.

Menu Tab Figure C-28: Example showing the **Switching > Link Aggregation** menu tab:

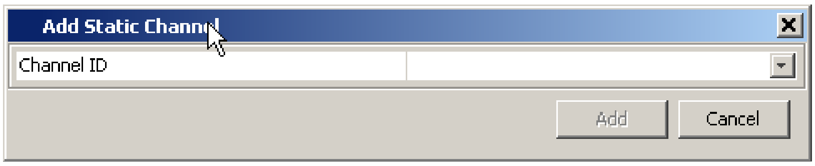

Allied Telesis
System
Switching
IP
Resiliency and High Availability
Management
Ports
VLANs
Link Aggregation
FDB Table
Power over Ethernet
x510-28GTX
Allied Telesis
awplus
Static Channels
Dynamic Channels (LACP)
Channel ID
Aggregate Partner
Add Dynamic Channel
Channel ID
Help
Add
Cancel
Add Dynamic Channel
Channel ID
Help
Add
Cancel
26/10/12 7:55 PM

Switching > Link Aggregation > Add Static Channel

The **Switching** > **Link Aggregation** > **Add Static Channel** dialog allows you to assign the selected port to a Static Channel group.

Configuration Dialog

Figure C-29: Example showing the **Switching > Link Aggregation > Add Static Channel** dialog:



Description

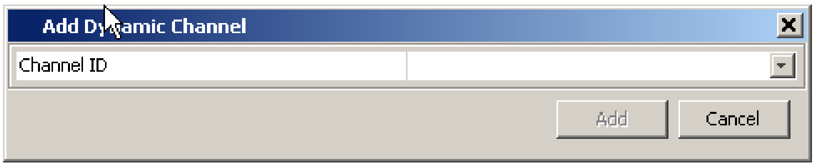
Label / Field / Button	Description
`Channel ID`	Specify a static channel group number for an interface. Up to 96 static channel groups can be created on the switch.

Switching > Link Aggregation > Add Dynamic Channel

The **Switching** > **Link Aggregation** > **Add Dynamic Channel** dialog allows you to assign the selected port to a Dynamic Channel (LACP) group.

Configuration Dialog

Figure C-30: Example showing **Switching > Link Aggregation > Add Dynamic Channel** dialog:



Description

Label / Field / Button	Description
`Channel ID`	Specify a dynamic (LACP) channel group number for an interface. Up to 32 dynamic (LACP) channel groups can be created on the switch.

Switching > FDB Table

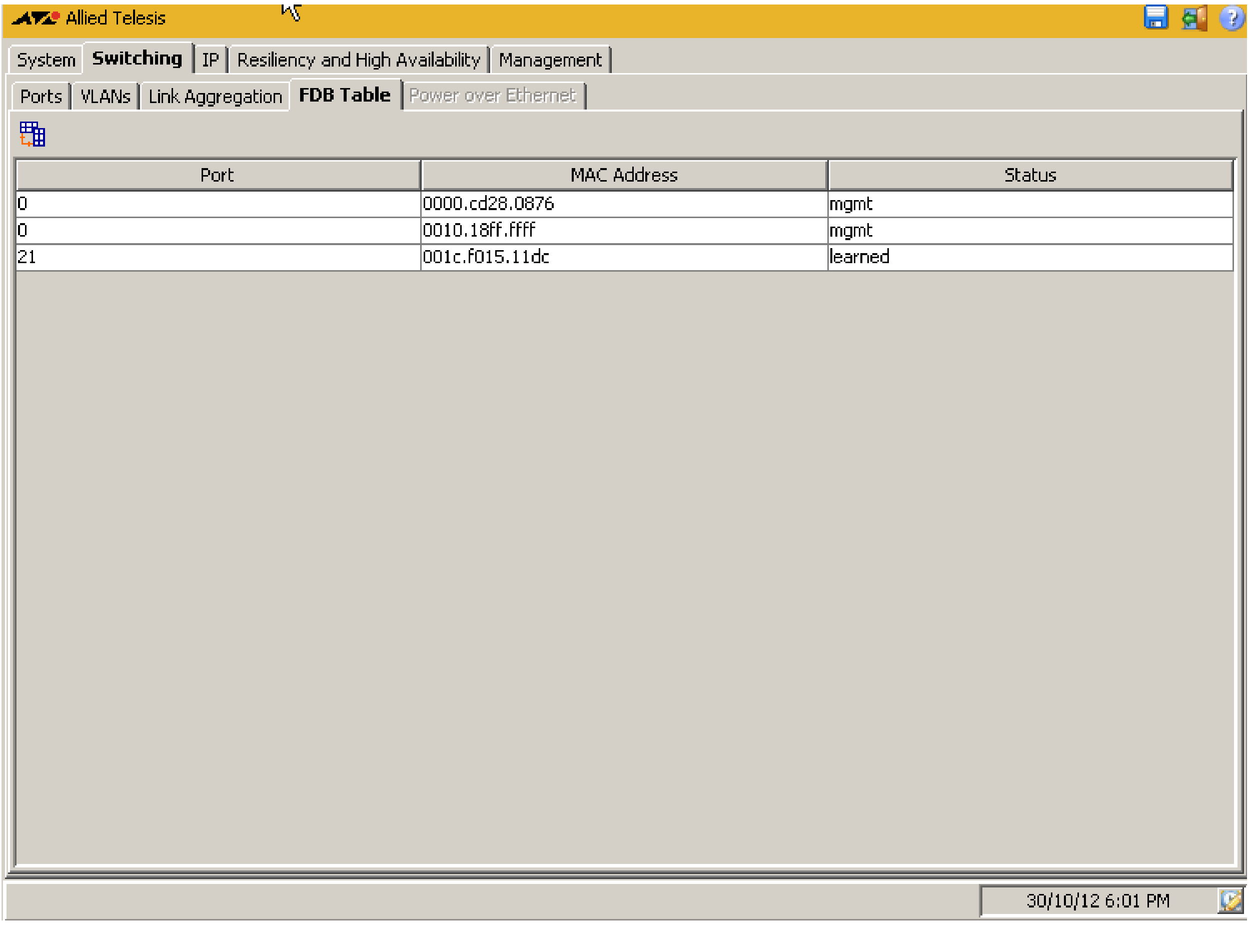
The **Switching > FDB Table** menu tab allows you to view the contents of the Layer 2 Forwarding Database Table.

You can change the FDB Table view to display horizontally or vertically by selecting the table view icon above the FDB Table.

You can also sort or rearrange the display of the FDB Table by Port, MAC Address, or Forwarding Status by selecting the relevant column or by dragging the relevant column respectively.

Menu Tab

Figure C-31: Example showing the **Switching > FDB Table** menu tab:



Port	MAC Address	Status
0	0000.cd28.0876	mgmt
0	0010.18ff.ffff	mgmt
21	001c.f015.11dc	learned

Description

Label / Field / Button	Description
`FDB Table`	Displays the FDB (Forwarding Database) table for the switch that shows all the available ports, MAC addresses, and port status.

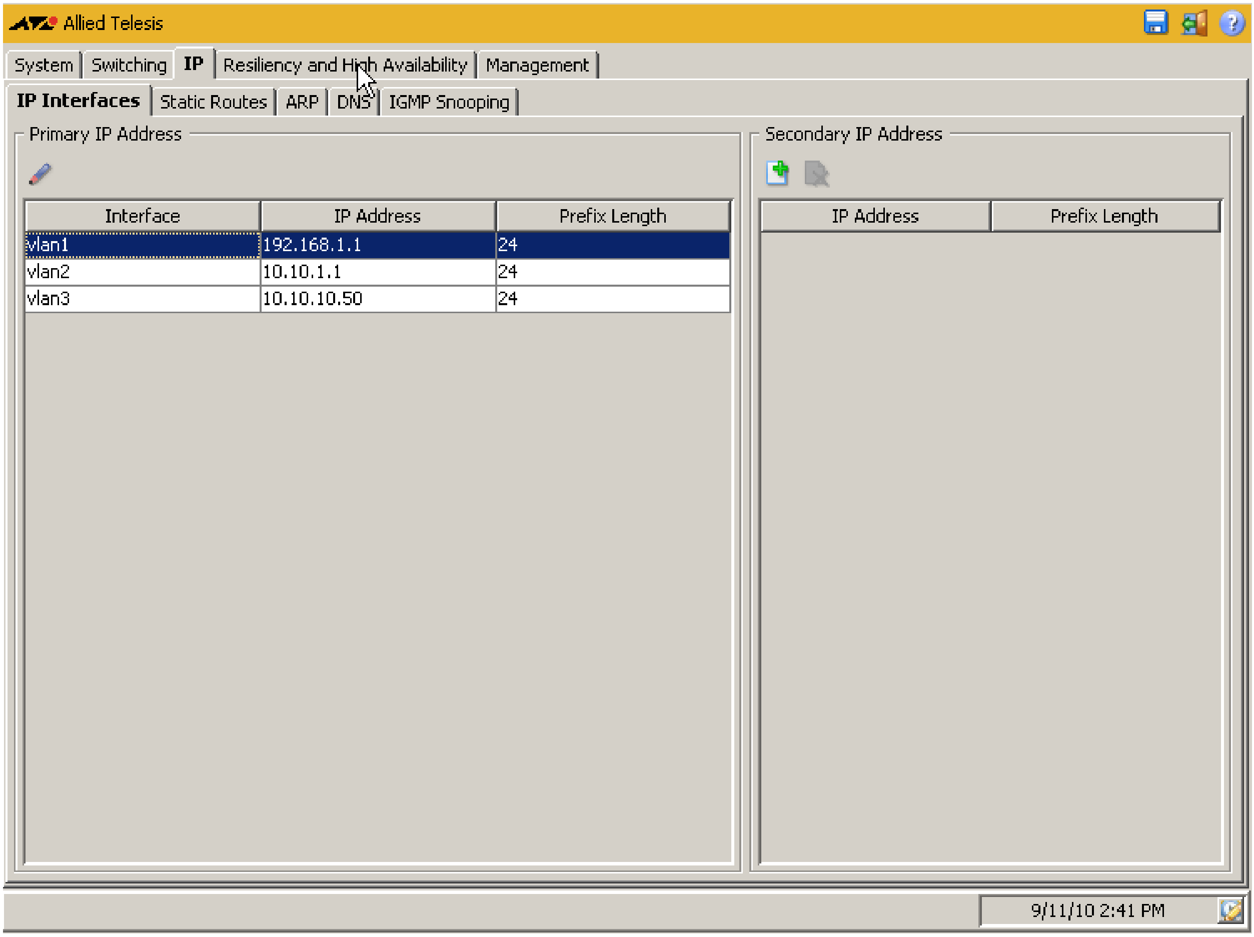
IP > IP Interfaces

The **IP > IP Interfaces** menu tab allows you to view and specify the Primary and Secondary IP Addresses for VLAN and management port interfaces.

Note You may only define a Secondary IP Address for an interface after first defining its Primary IP Address.

- Select an interface then use the pen shaped icon under the Primary IP Address label to configure an IP address. You can delete an assigned Primary IP Address in the Configure Primary IP Address dialog as displayed after selecting the icon under Primary IP Address.
- Select an interface with a Primary IP Address already defined to configure a Secondary IP Address, using the **+** icon under the Secondary IP Address label.
- Remove a selected Secondary IP Address using the **x** icon under the Secondary IP Address label.

Menu Tab Figure C-32: Example showing the **IP > IP Interfaces** menu tab:



Interface	IP Address	Prefix Length
vlan1	192.168.1.1	24
vlan2	10.10.1.1	24
vlan3	10.10.10.50	24

IP Address	Prefix Length

Description

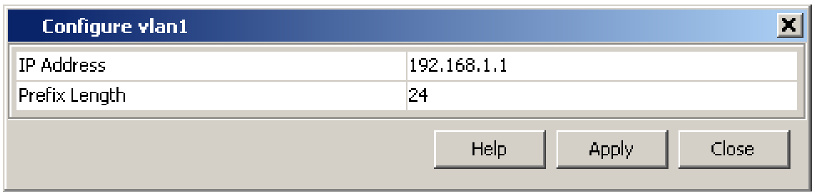
Label / Field / Button	Description
`Primary IP Address`	Displays and configures primary IP addressing for VLANs and management port interfaces that are defined on the switch and assigned to ports.
`Secondary IP Address`	Displays and configures secondary IP addressing for VLANs and management port interfaces that are defined on the switch and assigned to ports.

IP > IP Interfaces > Configure Primary IP Address

The **IP > IP Interfaces > Configure Primary IP Address** dialog allows you to configure a primary address with an IP address and a prefix length for the selected interface.

Configuration Dialog

Figure C-33: Example showing the **IP > IP Interfaces > Configure Primary IP Address** dialog:



Configure vlan1

IP Address	192.168.1.1
Prefix Length	24

Description

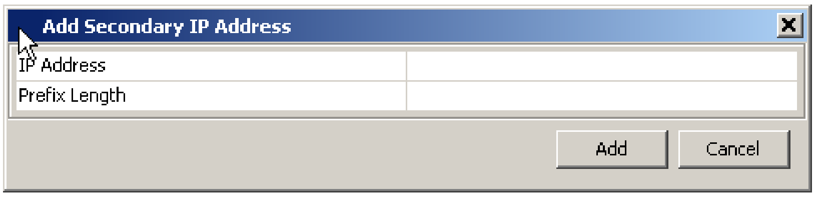
Label / Field / Button	Description
`IP Address`	Enter or remove an IPv4 Address in dotted decimal notation (i.e. A.B.C.D).
`Prefix Length`	Enter the Prefix for the IPv4 Address with the integer as used for slash notation (i.e. 24 instead of /24), not dotted decimal notation (i.e. 255.255.255.0).

IP > IP Interfaces > Add Secondary IP Address

The **IP > IP Interfaces > Add Secondary IP Address** dialog allows you to add a secondary address with an IP address and a prefix length for the selected interface (only if the selected interface already has a primary address configured).

Configuration Dialog

Figure C-34: Example showing the **IP > IP Interfaces > Add Secondary IP Address** dialog:



Add Secondary IP Address

IP Address	
Prefix Length	

Description

Label / Field / Button	Description
`IP Address`	Enter an IPv4 Address in dotted decimal notation (i.e. `A.B.C.D`).
`Prefix Length`	Enter the Prefix for the IPv4 Address with the integer as used for slash notation (i.e. `24` instead of `/24`), not dotted decimal notation (i.e. `255.255.255.0`).

IP > Static Routes

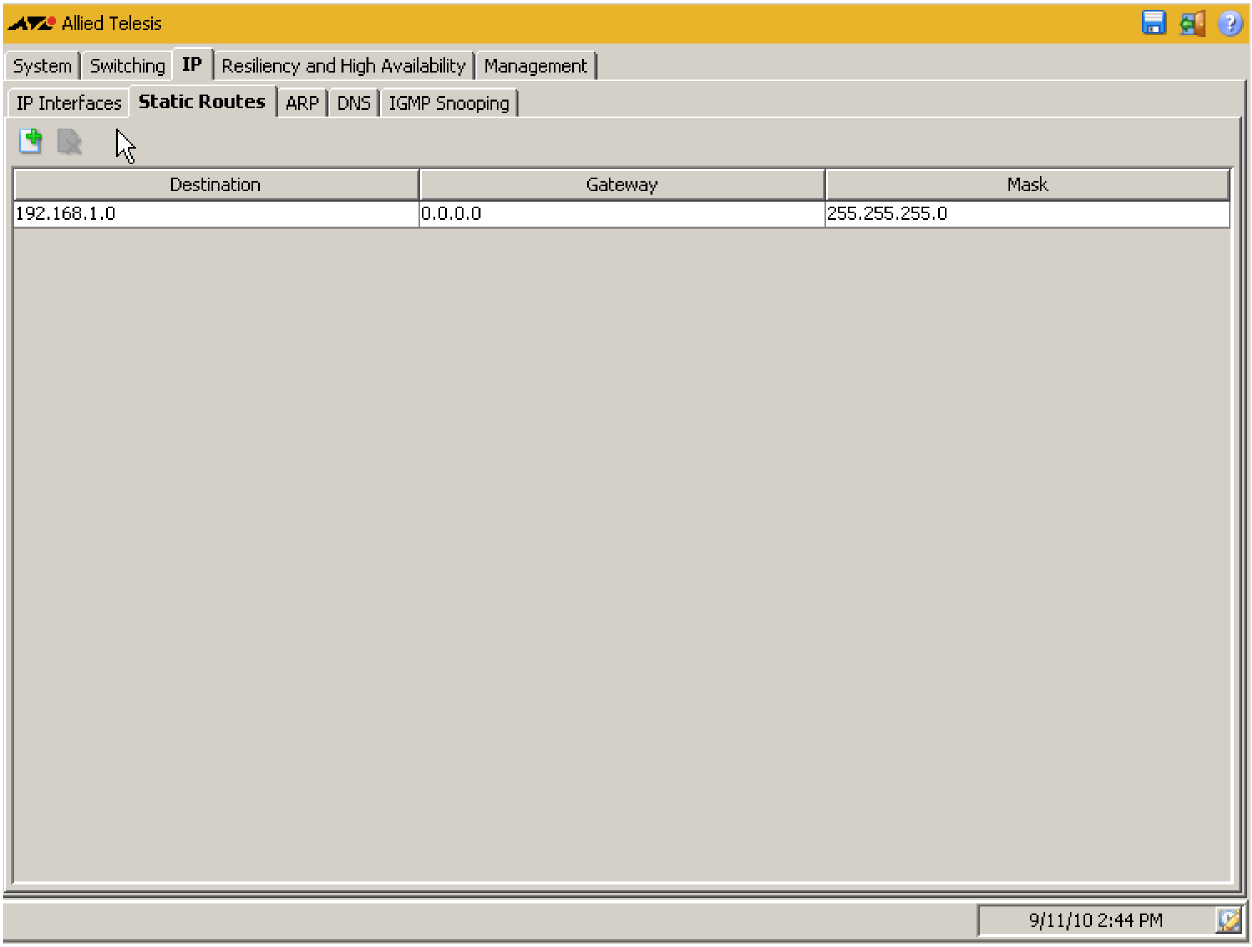
The **IP > Static Routes** menu tab allows you to view, add, and delete static IP routes configured on the switch. Static routes are specified using destination IP addresses, masks, and gateways.

You can also sort or rearrange the display of the Static Routes by Destination, Gateway, or Mask by selecting the relevant column or by dragging the relevant column respectively.

- Selecting the + icon allows you to define a Static Route specifying destination and gateway IPv4 addresses with a dotted decimal format subnet mask.
- Selecting the x icon allows you to delete a defined Static Route.

Menu Tab

Figure C-35: Example showing the **IP > Static Routes** menu tab:



Destination	Gateway	Mask
192.168.1.0	0.0.0.0	255.255.255.0

Description

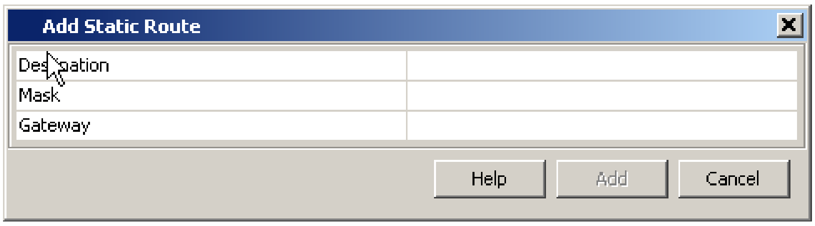
Label / Field / Button	Description
`Static Routes / Destination`	The IPv4 address of the destination subnet address.
`Static Routes / Gateway`	The IPv4 address of the gateway device.
`Static Routes / Mask`	The subnet mask in dotted decimal notation (for example, 255.255.255.0 instead of the slash notation /24).

IP > Static Routes > Add Static Route

The **IP > Static Routes > Add Static Route** dialog allows you to add a static IP routes on the switch. Static routes are specified using destination IP addresses, masks, and gateways.

Configuration Dialog

Figure C-36: Example showing the **IP > Static Routes > Add Static Route** dialog:



Description

Label / Field / Button	Description
`Destination`	Enter the IPv4 address of the destination subnet address.
`Mask`	Enter the subnet mask in dotted decimal notation (for example, 255.255.255.0 instead of the slash notation /24).
`Gateway`	Enter the IPv4 address of the gateway device.

IP > ARP

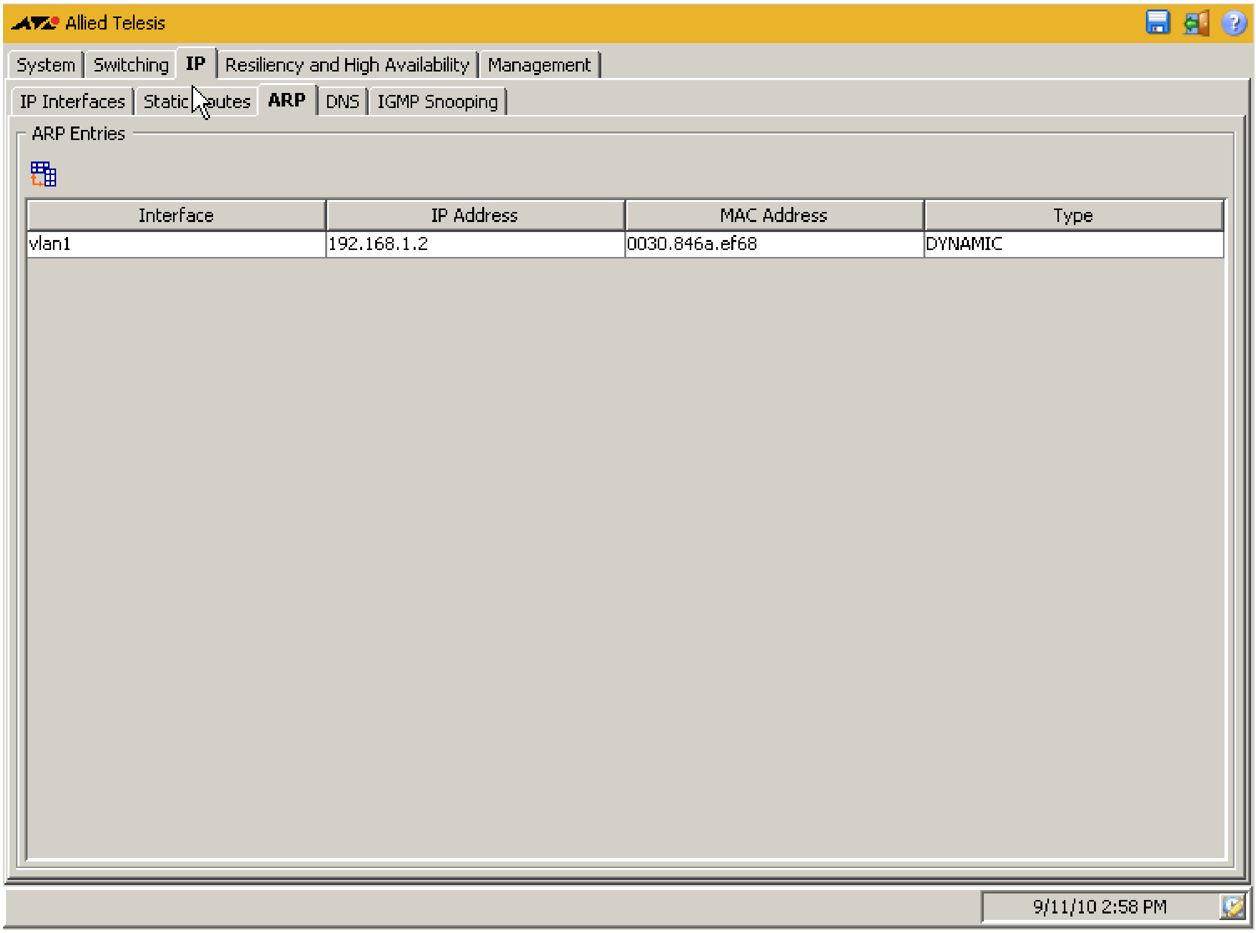
The **IP > ARP** menu tab allows you to view the contents of the ARP (Address Resolution Protocol) Table.

You can change the ARP Entries view to display horizontally or vertically by selecting the table view icon above the ARP Entries.

You can also sort or rearrange the display of the ARP Entries by Interface, IP Address, MAC Address, or Type by selecting the relevant column or by dragging the relevant column respectively.

Menu Tab

Figure C-37: Example showing the **IP > ARP** menu tab:



Interface	IP Address	MAC Address	Type
vlan1	192.168.1.2	0030.846a.ef68	DYNAMIC

Description

Label / Field / Button	Description
`ARP Entries / Interface`	Interface over which the switch is accessed, usually a VLAN.
`ARP Entries / IP Address`	IP address of the network device this ARP entry maps to.

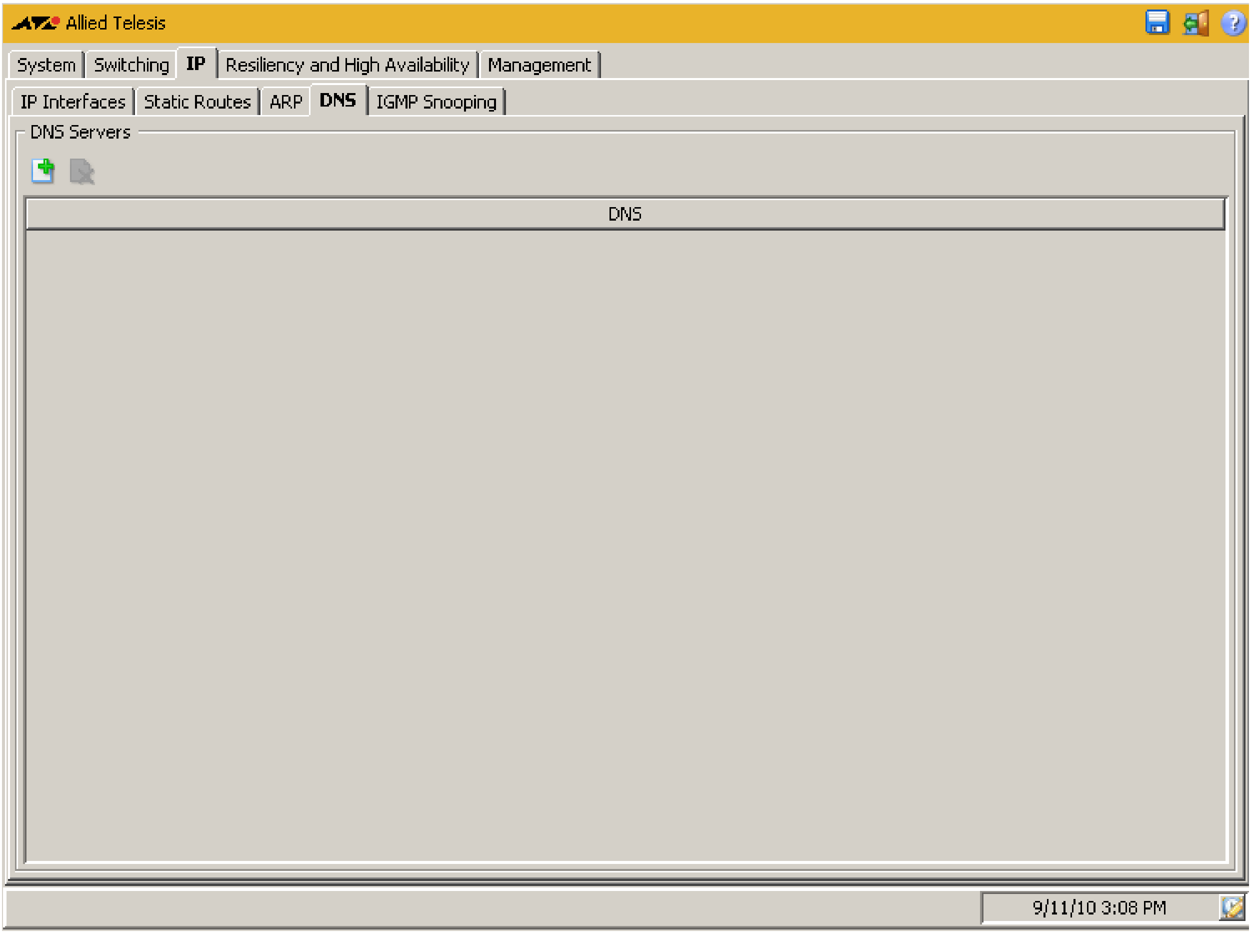
Label / Field / Button	**Description**(cont.)
`ARP Entries / MAC Address`	Hardware address of the switch in hexadecimal format `HHHH.HHHH.HHHH`.
`ARP Entries / Type`	Indicates whether the ARP entry is a Static or Dynamic ARP entry. Static ARP entries are added and dynamic ARP entries are learned.

IP > DNS

The **IP > DNS** menu tab allows you to display and configure DNS (Domain Name System) server entries for the switch.

- Selecting the + icon allows you to define a DNS Server specifying the IPv4 address.
- Selecting the x icon allows you to delete a defined DNS Server.

Figure C-38: Example showing the **IP > DNS** menu tab:



Description

Label / Field / Button	Description
DNS	DNS Server IPv4 address.

IP > DNS > Add DNS Server

The **IP > DNS > Add DNS Server** dialog allows you to add DNS (Domain Name System) server entries for the switch.

Configuration Dialog

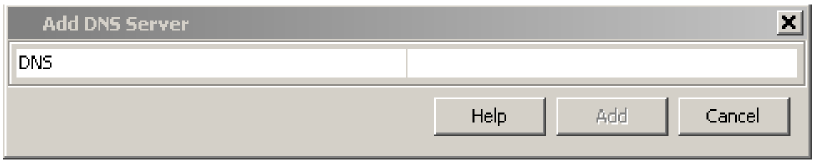


Figure C-39: Example showing the **IP > DNS > Add DNS Server** dialog:

Description

Label / Field / Button	Description
DNS	Enter an IPv4 address in dotted decimal notation (i.e. A.B.C.D) for the DNS (Domain Name System) Server you want to specify.

IP > IGMP Snooping

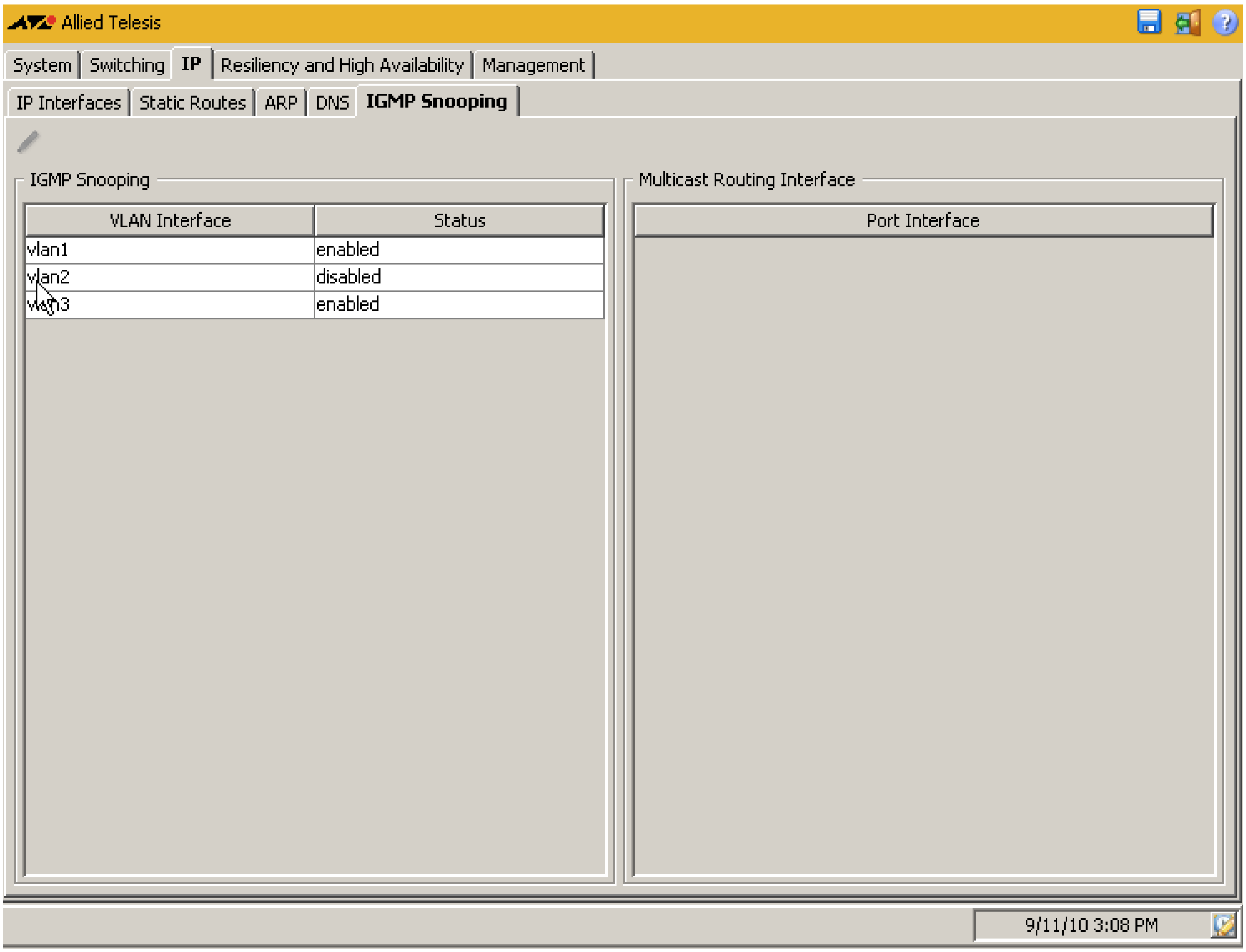
Menu Tab

The **IP > IGMP Snooping** menu tab displays basic IGMP Snooping and Multicast Routing Interface information.

You can also configure IGMP Snooping on individual ports by selecting the VLAN interface that the port is a member of then clicking on the pen shaped icon to display the Configure IGMP Snooping dialog, where you can enable or disable IGMP snooping on desired ports.

Menu Tab

Figure C-40: Example showing the **IP > IGMP Snooping** menu tab:



VLAN Interface	Status
vlan1	enabled
vlan2	disabled
vlan3	enabled

Description

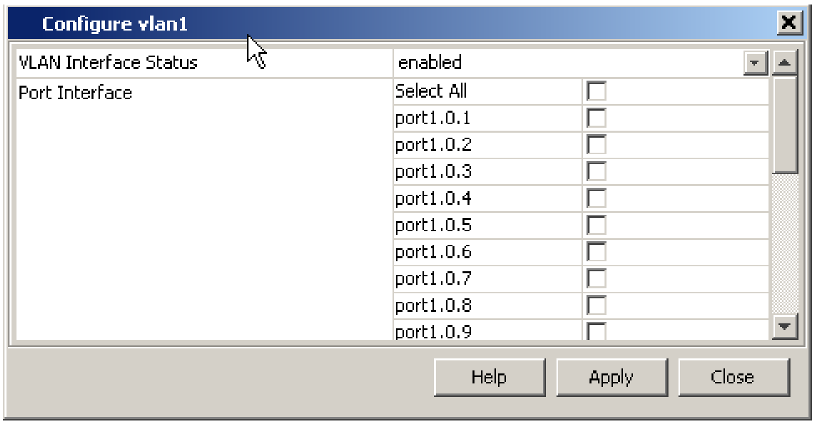
Label / Field / Button	Description
`IGMP Snooping / IGMP Snooping`	Displays and configures IGMP Snooping for a specified VLAN interface.
`IGMP Snooping / Multicast Routing Interface`	Displays and configures the specified port in the VLAN as a multicast router for IGMP Snooping.

IP > IGMP Snooping > Configure Interface

The **IP > IGMP Snooping > Configure Interface** dialog allows you to configure IGMP Snooping on individual ports. First select the VLAN interface that the port is a member of then enable or disable IGMP snooping on desired ports from this dialog.

Configuration Dialog

Figure C-41: Example showing the **IP > IGMP Snooping > Configure Interface** dialog:



Configure vlan1

VLAN Interface Status	enabled	
Port Interface	Select All	□
	port1.0.1	□
	port1.0.2	□
	port1.0.3	□
	port1.0.4	□
	port1.0.5	□
	port1.0.6	□
	port1.0.7	□
	port1.0.8	□
	port1.0.9	□

Description

Label / Field / Button	Description
`VLAN Interface Status`	Displays whether the selected VLAN is enabled or disabled.
`Port Interface`	Select the check box for a given port to allocate it to the VLAN.

Resiliency and High Availability > STP

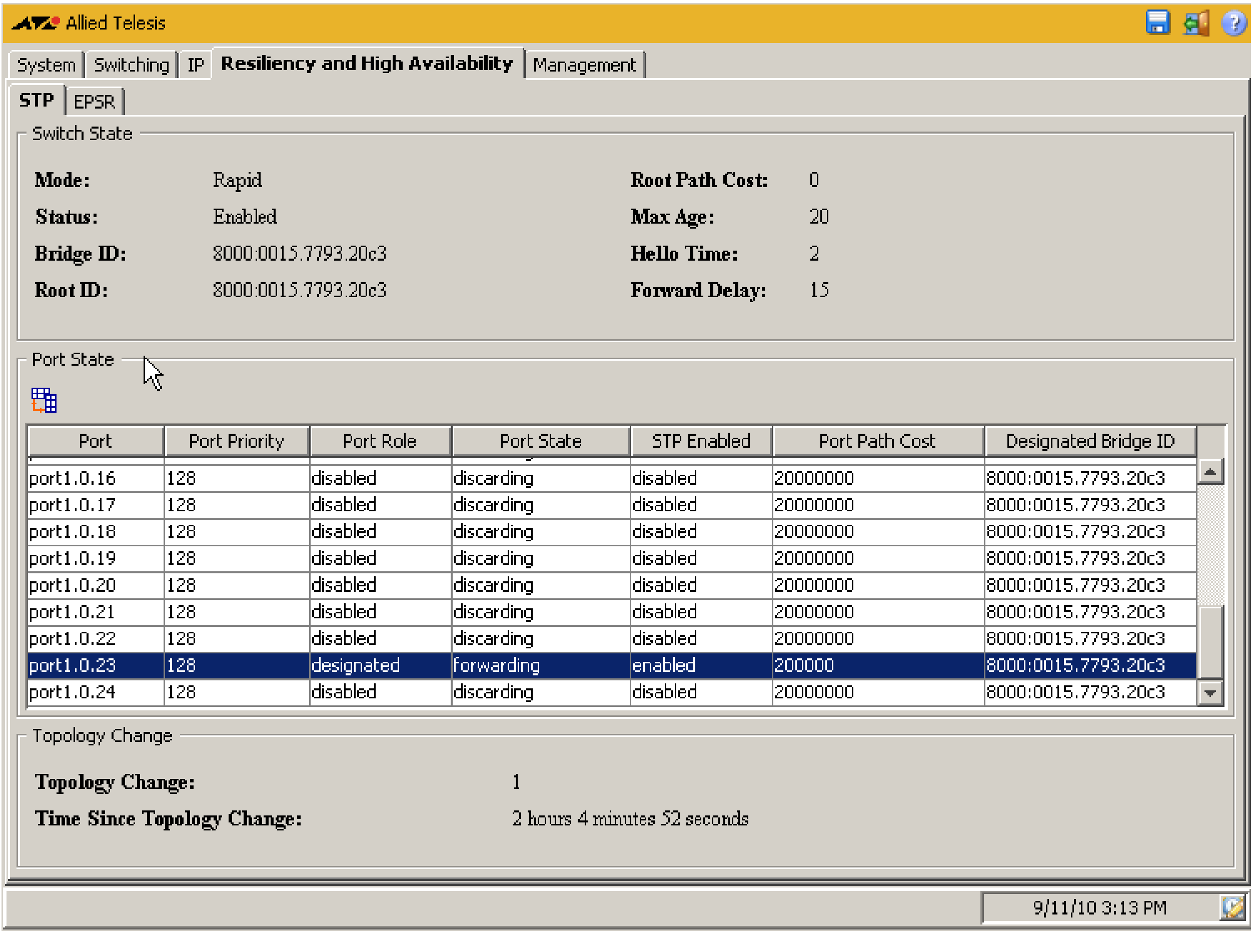
The **Resiliency and High Availability > STP** menu tab allows you to view the configuration and status of spanning tree data: for the switch as a whole and for each port.

You can also sort or rearrange the display of the Port State table by Port, Port Priority, Port State, Port Role, STP Enabled, Port Path Cost, or Designated Bridge ID by selecting the relevant column or by dragging the relevant column respectively.

Note STP is not configurable through the GUI. Refer to the relevant STP chapters in the AlliedWare Plus^{TM} Software Reference to configure STP using the CLI instead.

Menu Tab

Figure C-42: Example showing the **Resiliency and High Availability > STP** menu tab:



Mode: Rapid

Status: Enabled

Bridge ID: 8000:0015.7793.20c3

Root ID: 8000:0015.7793.20c3

Root Path Cost: 0

Max Age: 20

Hello Time: 2

Forward Delay: 15

Port	Port Priority	Port Role	Port State	STP Enabled	Port Path Cost	Designated Bridge ID
port1.0.16	128	disabled	discarding	disabled	20000000	8000:0015.7793.20c3
port1.0.17	128	disabled	discarding	disabled	20000000	8000:0015.7793.20c3
port1.0.18	128	disabled	discarding	disabled	20000000	8000:0015.7793.20c3
port1.0.19	128	disabled	discarding	disabled	20000000	8000:0015.7793.20c3
port1.0.20	128	disabled	discarding	disabled	20000000	8000:0015.7793.20c3
port1.0.21	128	disabled	discarding	disabled	20000000	8000:0015.7793.20c3
port1.0.22	128	disabled	discarding	disabled	20000000	8000:0015.7793.20c3
port1.0.23	128	designated	forwarding	enabled	200000	8000:0015.7793.20c3
port1.0.24	128	disabled	discarding	disabled	20000000	8000:0015.7793.20c3

Topology Change: 1

Time Since Topology Change: 2 hours 4 minutes 52 seconds

Description: Switch State

Label / Field / Button	Description
`Switch State / Mode`	Spanning Tree Mode displayed: STP (Spanning Tree Protocol), Rapid (Rapid Spanning Tree Protocol - RSTP), or Multiple (Multiple Spanning Tree Protocol - MSTP).
`Switch State / Status`	Status of the Spanning Tree Mode: enabled or disabled.
`Switch State / Bridge ID`	Bridge ID, comprising the port priority followed by its MAC address.
`Switch State / Root ID`	Root Bridge ID, comprising the root priority followed by its MAC address.
`Switch State / Root Path Cost`	Sum of the costs for each path between the bridge port and the root bridge.
`Switch State / Max Age`	Time in seconds that the dynamic spanning tree configuration information is stored in the switch before it is discarded.
`Switch State / Hello Time`	Time in seconds between the transmission of switch spanning tree configuration information, when the switch is the Root Bridge of the spanning tree or is trying to become the Root Bridge.
`Switch State / Forward Delay`	Time in seconds to control how fast a port changes its spanning tree state when moving towards the forwarding state. This value is used only when the switch is acting as the root bridge. Note that Forward Delay, Max Age, and Hello Time are interrelated.

Description: Port State

Label / Field / Button	Description
`Port State / Port`	Switch port number in the format 'portX.Y.Z' where X is the switch, Y is the XEM, and Z is the individual switch port number.
`Port State / Port Priority`	The lower the port priority, the higher the likelihood of the port becoming part of the active network topology.
`Port State / Port Role`	Displays the port role as configured in the CLI with 'spanning-tree' commands, and shows either 'rootport', 'backup', 'disabled' or 'designated' port roles.
`Port State / Port State`	Displays the spanning tree state for the port as configured in the CLI with 'spanning-tree' commands. Indicates spanning tree states of: disabled, blocking, listening, learning, and forwarding.
`Port State / STP Enabled`	Displays whether spanning-tree is enabled or disabled. Spanning tree is enabled by default.
`Port State / Port Path Cost`	The cost of a path for the port that determines the total cost path. The lower the total cost, the higher the priority of the path.
`Port State / Designated Bridge ID`	The unique parent for each bridge that connects it to the next LAN on the path towards the root bridge.

Description: Topology Change

Label / Field / Button	Description
`Topology Change / Topology Change`	The number of STP Topology Changes that have occurred since the switch was rebooted.
`Topology Change / Time Since Topology Change`	The time in hours and seconds since the previous STP Topology Change occurred.

Resiliency and High Availability > EPSR

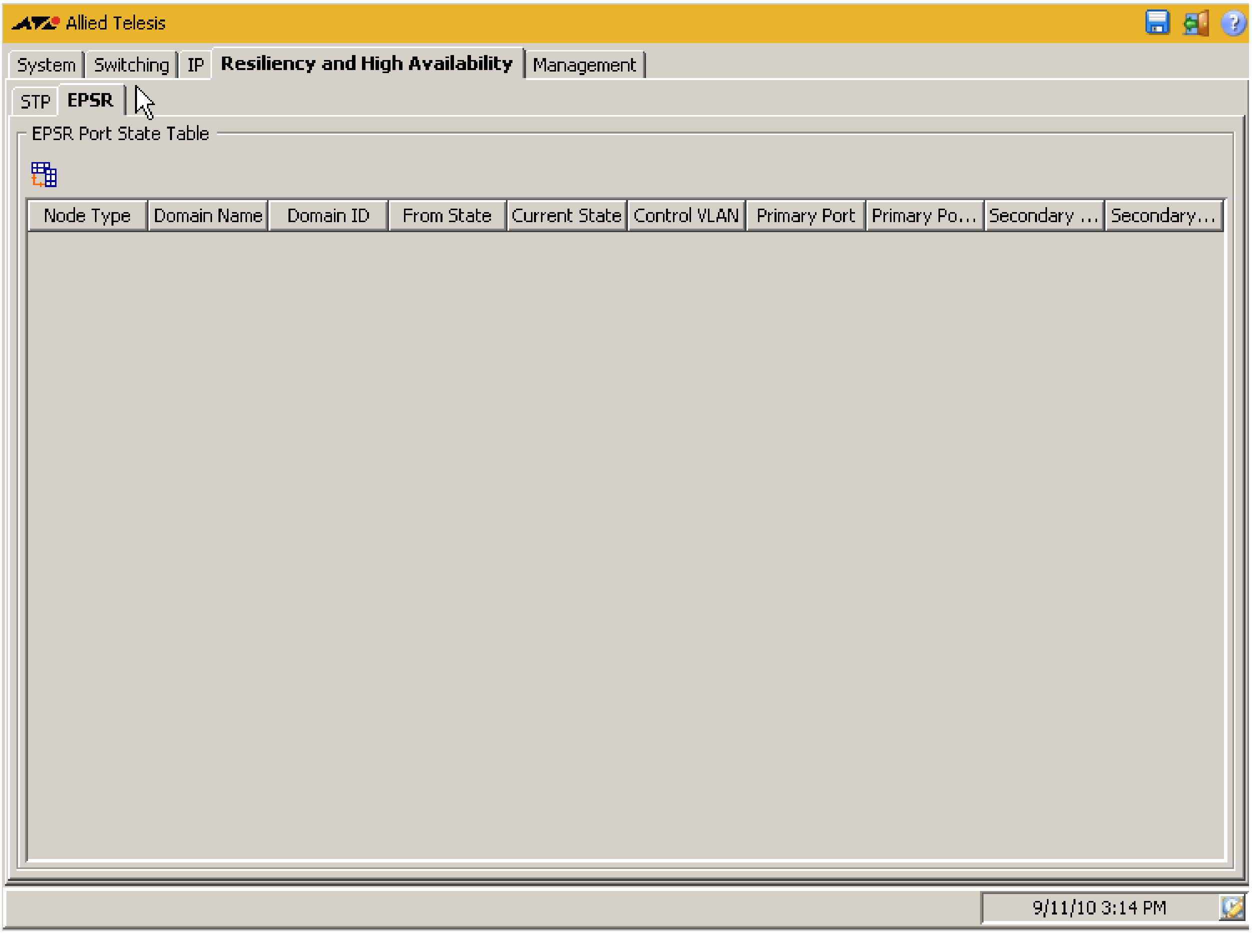
The **Resiliency and High Availability > EPSR** menu tab allows you to display the properties and status of any EPSR domains configured on the switch.

You can also sort or rearrange the display of the EPSR Port State table by Node Type, Domain Name, Domain ID, From State, Current State, Control VLAN, Primary Port, Primary Port Status, Secondary Port, or Secondary Port Status by selecting the relevant column or by dragging the relevant column respectively.

Note EPSR is not configurable through the GUI. Refer to the relevant EPSR chapters in the AlliedWare Plus^{TM} Software Reference to configure EPSR using the CLI instead.

Menu Tab

Figure C-43: Example showing the **Resiliency and High Availability > EPSR** menu tab:



Node Type	Domain Name	Domain ID	From State	Current State	Control VLAN	Primary Port	Primary Po...	Secondary ...	Secondary...

9/11/10 3:14 PM

Description

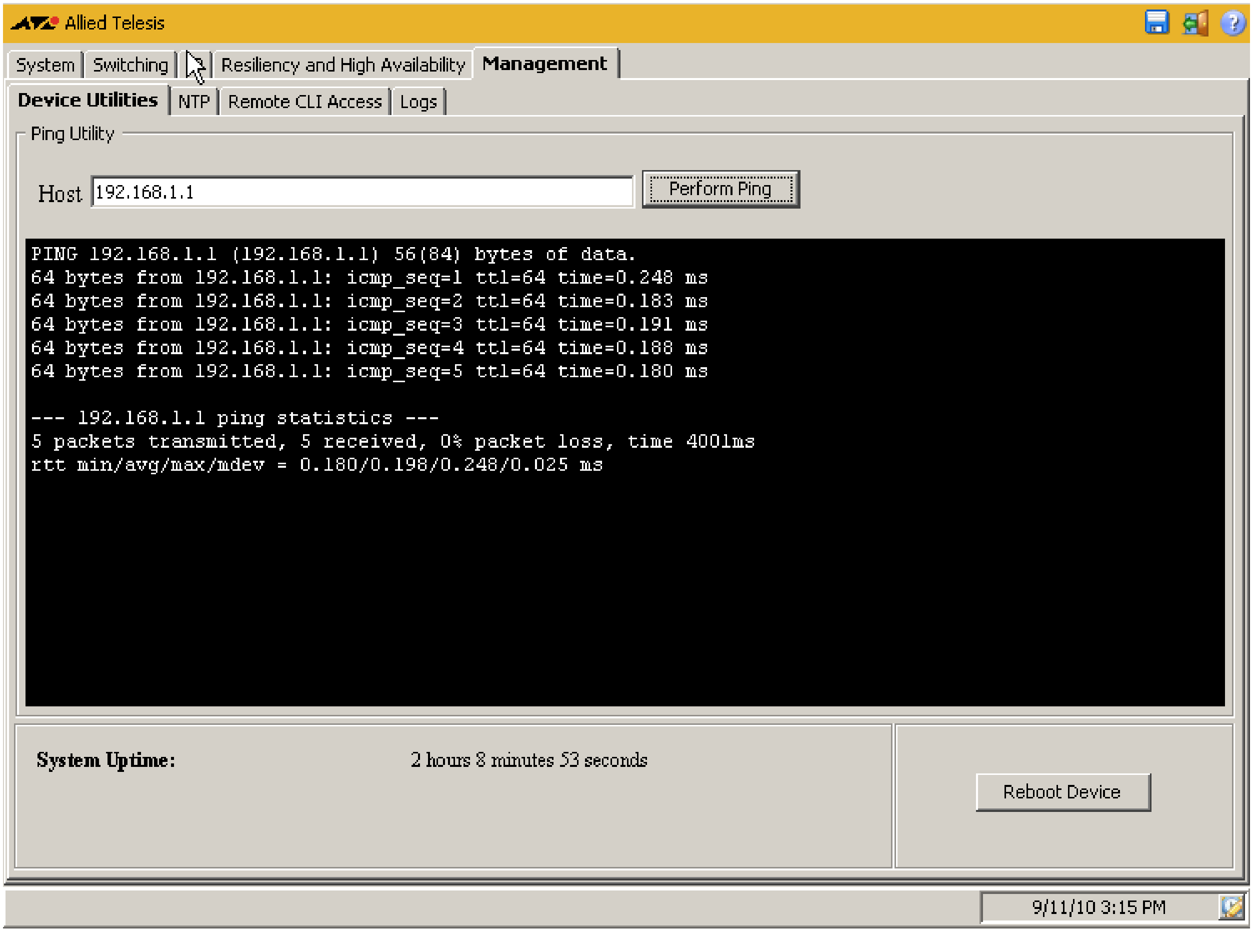
Label / Field / Button	Description
`EPSR Port State Table / Node Type`	Displays master or transit node as configured in the CLI with the '**epsr mode**' command.
`EPSR Port State Table / Domain Name`	Displays the domain name. A set of instances across a ring is called a domain.
`EPSR Port State Table / Domain ID`	Displays the assigned domain number for the domain name.
`EPSR Port State Table / From State`	Displays the From EPSR state as configured in the CLI with the '**epsr state**' commands.
`EPSR Port State Table / Current State`	Displays the Current EPSR state as configured in the CLI with the '**epsr state**' commands.
`EPSR Port State Table / Control VLAN`	Displays the control VLAN as configured in the CLI with the '**epsr mode controlvlan**' command.
`EPSR Port State Table / Primary Port`	Displays the master node primary port interface name as configured in the CLI with the '**epsr mode primaryport**' command.
`EPSR Port State Table / Primary Port Status`	Displays the master node primary port interface status: up or down.
`EPSR Port State Table / Secondary Port`	Displays the assigned secondary port interface name.
`EPSR Port State Table / Secondary Port Status`	Displays the assigned secondary port interface status: up or down.

Management > Device Utilities

The **Management > Device Utilities** menu tab allows you to perform pings and reboot the switch from the GUI.

Menu Tab

Figure C-44: Example showing the **Management > Device Utilities** menu tab:



```
PING 192.168.1.1 (192.168.1.1) 56(84) bytes of data.
64 bytes from 192.168.1.1: icmp_seq=1 ttl=64 time=0.248 ms
64 bytes from 192.168.1.1: icmp_seq=2 ttl=64 time=0.183 ms
64 bytes from 192.168.1.1: icmp_seq=3 ttl=64 time=0.191 ms
64 bytes from 192.168.1.1: icmp_seq=4 ttl=64 time=0.188 ms
64 bytes from 192.168.1.1: icmp_seq=5 ttl=64 time=0.180 ms

--- 192.168.1.1 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4001ms
rtt min/avg/max/mdev = 0.180/0.198/0.248/0.025 ms
```

System Uptime: 2 hours 8 minutes 53 seconds

Description

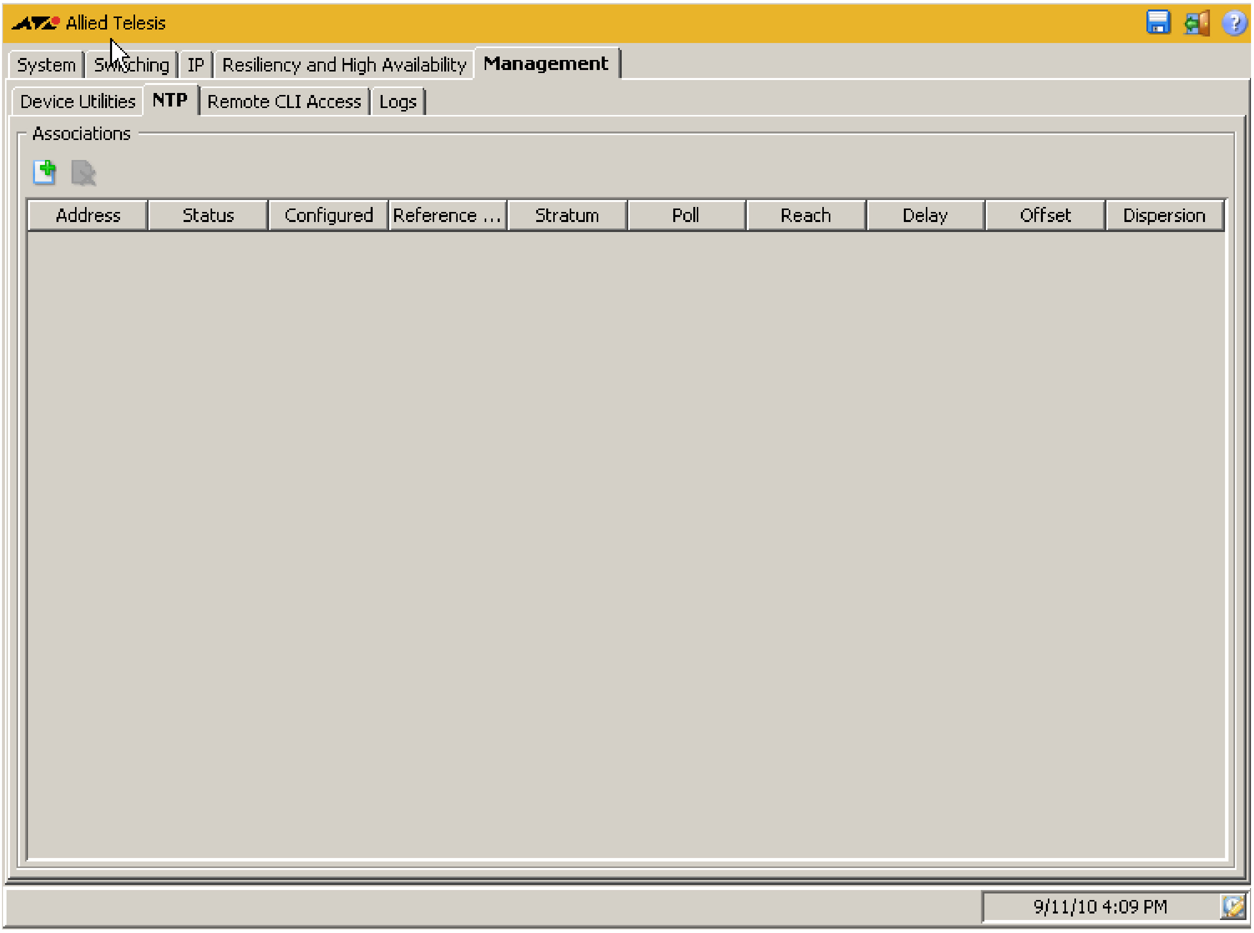
Label / Field / Button	Description
`Ping Utility / Host`	Enter the IPv4 address or the URL that you want to ping in this field.
`Ping Utility / Perform Ping`	Select this button to ping the IPv4 address or URL that you entered in the Host field.
`System Uptime`	Displays the elapsed time since the last reboot in hours, minutes, and seconds.
`Reboot Device`	Select this button to reboot your switch. You will need to login to the GUI again after you reboot your switch. Rebooting closes all Telnet / SSH / SNMP sessions on your switch.

Management > NTP

The **Management > NTP** menu tab allows you to display and configure Network Time Protocol (NTP) peer configurations on the switch.

Menu Tab

Figure C-45: Example showing the **Management > NTP** menu tab:



Description

Label / Field / Button	**Description**
`Associations / Address`	The NTP peer or NTP server IPv4 address.
`Associations / Status`	Indicates association status, and displays '**master(synced)**'. '**master(unsynced)**', '**selected**', '**candidate**', '**configured**', or '**unknown**'.
`Associations / Configured`	Indicates if the association is configured or not, and displays '**configured**' or '**dynamic**'.
`Associations / Reference Clock`	The IPv4 address for the reference clock.

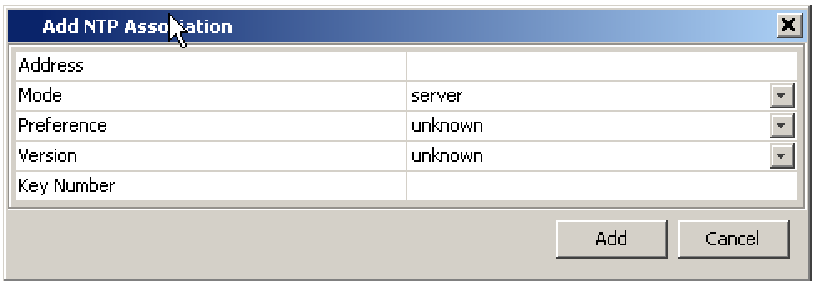
Label / Field / Button(cont.)	**Description**(cont.)
Associations / Stratum	The number of hops between the server and the accurate time source.
Associations / Poll	The time between NTP requests from the device to the server.
Associations / Reach	Shows whether or not the NTP server responded to the last request, which indicates the reachability of the NTP peer.
Associations / Delay	The round trip delay between the device and the server.
Associations / Offset	The difference between the device clock and the server clock, relative to the server clock, in milliseconds.
Associations / Dispersion	The lowest measure of error associated with peer offset based on delay.

Management > NTP > Add NTP Association

The **Management > NTP > Add NTP Association** dialog allows you configure Network Time Protocol (NTP) peer configurations on the switch.

Configuration Dialog

Figure C-46: Example showing the **Management > NTP > Add NTP Association** dialog:



Add NTP Association

Address	
Mode	server
Preference	unknown
Version	unknown
Key Number	

Description

Label / Field / Button	**Description**
Address	Enter the NTP IPv4 address for the NTP peer or NTP server used.
Mode	Select one of the server or peer options from the drop down list to specify the NTP Mode used. When using NTP server mode, the NTP server will not accept updates from clients for updates to the server's time settings. The NTP server is configured to synchronize the NTP clients. When using NTP peer mode, each device shares its time information with the other, and each device can also provide time synchronization to the other.

Label / Field / Button(cont.)	**Description**(cont.)
`Preference`	Select one of the `unknown`, `not preferred`, or `preferred` options from the drop down list to specify the NTP Preference used. NTP Preference is used to configure an NTP server, so the NTP server is given preference to synchronize the NTP clients.
`Version`	Select one of the `unknown`, `version 1`, `version 2`, `version 3`, or `version 3, version 4` options from the drop down list to specify the NTP Version used.
`Key Number`	Enter the NTP Key Number for NTP authentication, which allows NTP to authenticate the associations with other systems for security purposes. The NTP Key Number is an integer in the range <1-4294967295>. The MD5 (Message-Digest algorithm 5) key type is supported to encrypt the NTP Key Number used for authentication.

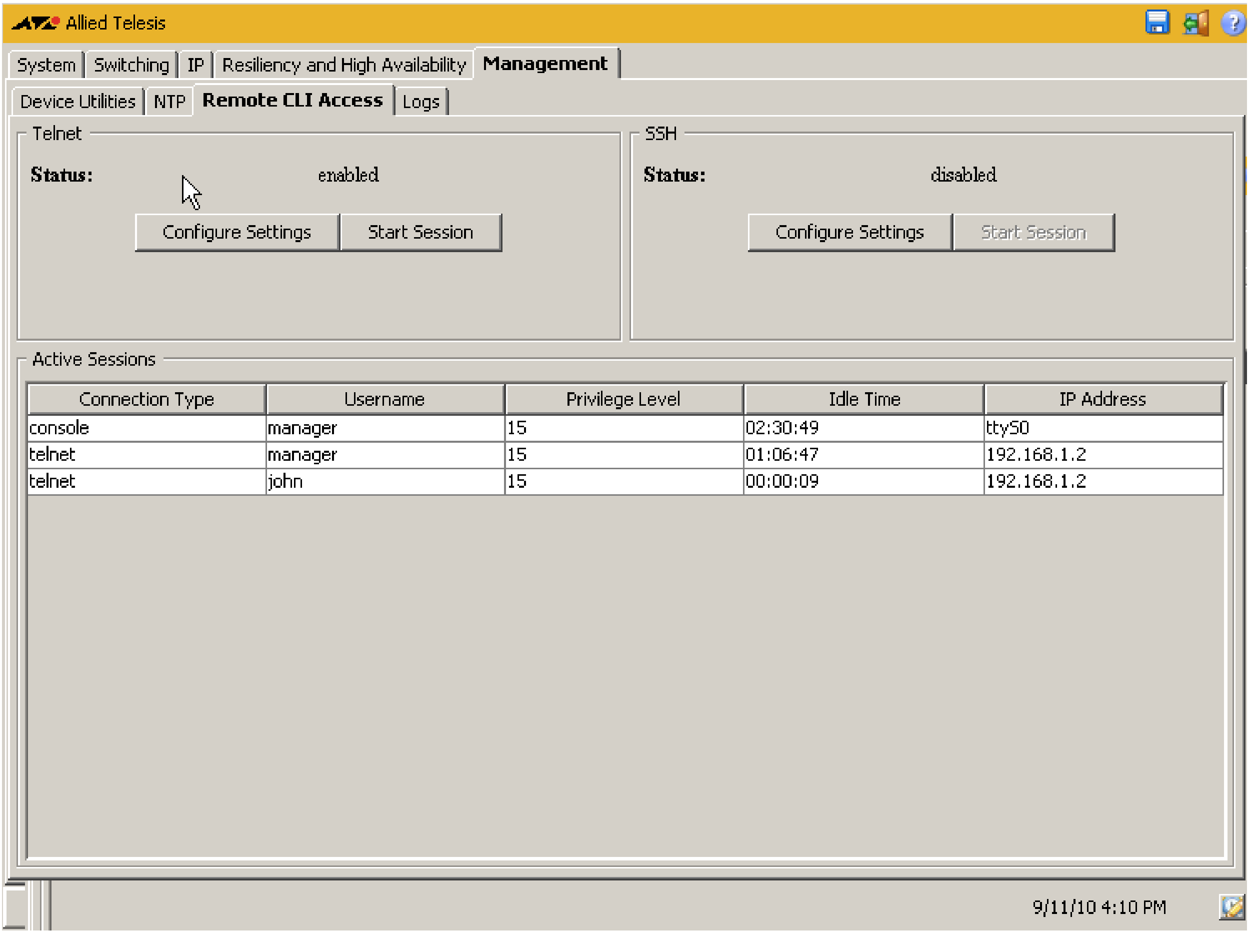
Management > Remote CLI Access

The **Management > Remote CLI Access** menu tab allows you to enable, disable and configure Telnet and SSH.

You can create Telnet or SSH connections to the switch, and you can view a list of all current active CLI sessions on the switch from this tab.

Menu Tab

Figure C-47: Example showing the **Management > Remote CLI Access** menu tab:



Telnet

Status: enabled

SSH

Status: disabled

Active Sessions

Connection Type	Username	Privilege Level	Idle Time	IP Address
console	manager	15	02:30:49	ttyS0
telnet	manager	15	01:06:47	192.168.1.2
telnet	john	15	00:00:09	192.168.1.2

9/11/10 4:10 PM

Description

Label / Field / Button	Description
`Telnet / Status`	Displays the current Telnet status, either ‘`enabled`’ or “`disabled`’.
`Telnet / Configure Settings`	Configures the Telnet Status. Select ‘`enabled`’ or ‘`disabled`’ to configure the status of the Telnet server on the switch.
`Telnet / Start Session`	Starts a Telnet session to use the CLI. After starting a Telnet session you will need to login to the switch to use the CLI.
`SSH / Status`	Displays the current SSH status, either ‘`enabled`’ or ‘`disabled`’.
`SSH / Configure Settings`	Configures the SSH Status. Select ‘`enabled`’ or ‘`disabled`’ to configure the status of the SSH server on the switch. Note that relevant certificates must be installed to initiate an SSH session.
`SSH / Start Session`	Starts a secure SSH session to use the CLI. After starting an SSH session you will need to login to the switch to use the CLI.

Description

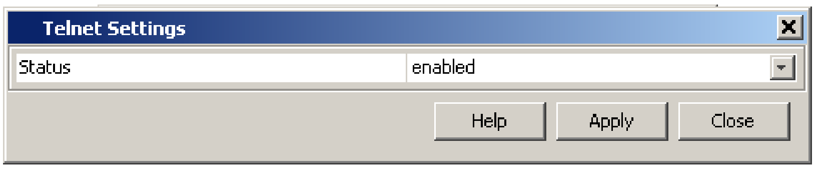
Label / Field / Button	Description
`Active Sessions / Connection Type`	A Console connection or a VTY connection.
`Active Sessions / Username`	Login name for a user.
`Active Sessions / Privilege Level`	The privilege set for a user for VTY or console connection. Privilege levels range from 0-15 with 15 the highest privilege level. Privilege levels are used in the CLI to enable or disable access to different configuration modes and commands. Privilege levels 0-14 only enables users to view system configuration and system behavior. Privilege level 15 enables users to globally configure all the interfaces on a switch.
`Active Sessions / Idle Time`	Time in seconds that the SSH Server waits to receive data from the SSH Client. The SSH Server disconnects when the Idle Time limit is reached.
`Active Sessions / IP Address`	The IPv4 address for the VTY connection.

Management > Remote CLI Access > Telnet Settings

The **Management > Remote CLI Access > Telnet Settings** dialog allows you to enable or disable Telnet.

Configuration Dialog

Figure C-48: Example showing **Management > Remote CLI Access > Telnet Settings** dialog:


Telnet Settings

Status: enabled

Description

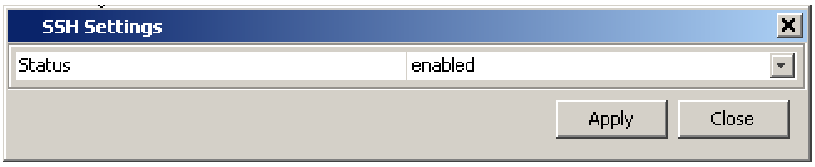
Label / Field / Button	Description
`Status`	Select `enabled` or `disabled` from the drop down list on this dialog to enable or disable Telnet respectively on the switch.

Management > Remote CLI Access > SSH Settings

The **Management > Remote CLI Access > SSH Settings** dialog allows you to enable or disable SSH.

Configuration Dialog

Figure C-49: Example showing the **Management > Remote CLI Access > SSH Settings** dialog:


SSH Settings

Status: enabled

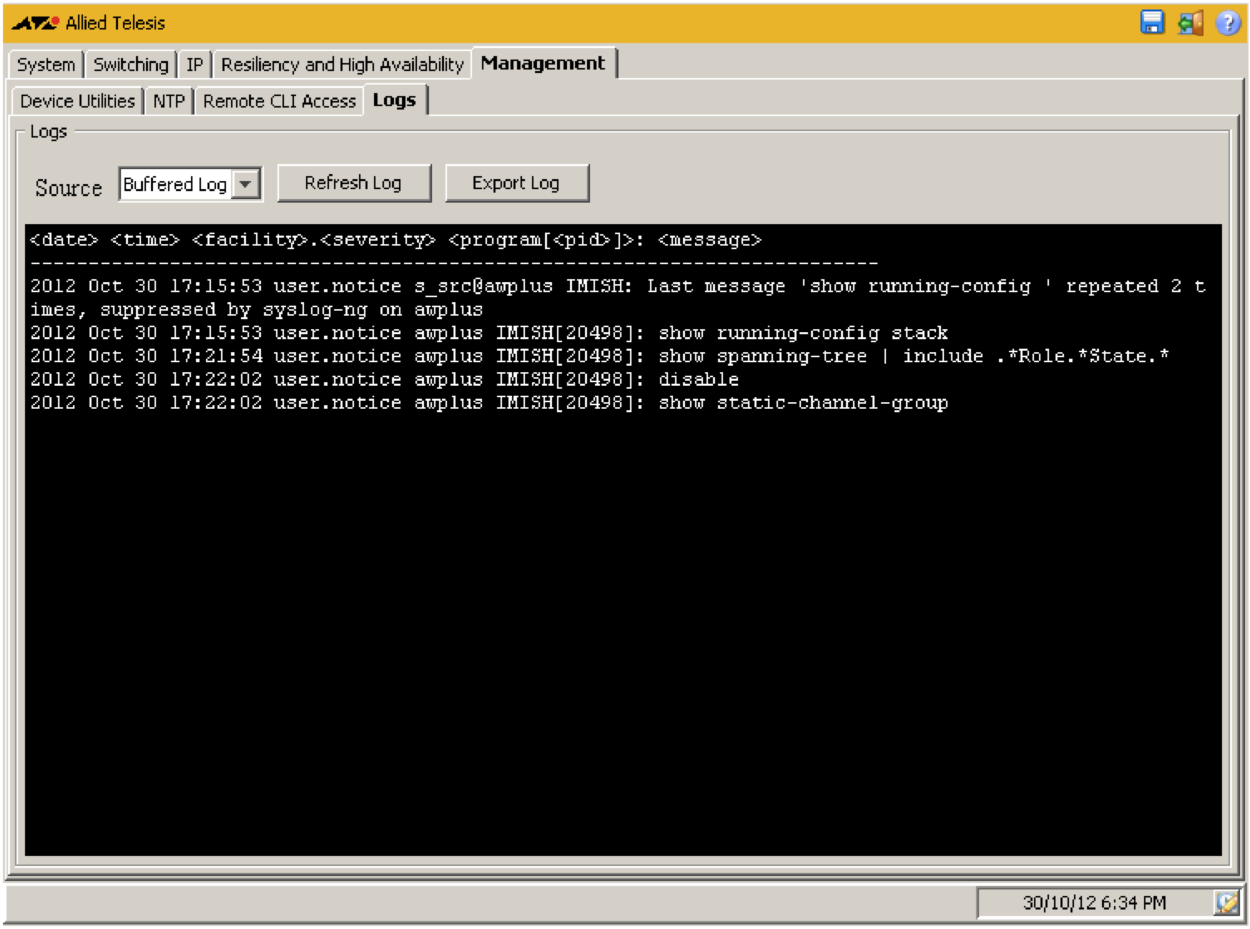
Description

Label / Field / Button	Description
`Status`	Select `enabled` or `disabled` from the drop down list on this dialog to enable or disable SSH respectively on the switch.

Management > Logs

The **Management >Logs** menu tab allows you to view the switch logs, and export the switch logs as .csv format files.

Menu Tab Figure C-50: Example showing the **Management > Logs** menu tab:



Source: Buffered Log

```
<date> <time> <facility>.<severity> <program[<pid>]>: <message>
------------------------------------------------------------------------
2012 Oct 30 17:15:53 user.notice s_src@awplus IMISH: Last message 'show running-config ' repeated 2 t
imes, suppressed by syslog-ng on awplus
2012 Oct 30 17:15:53 user.notice awplus IMISH[20498]: show running-config stack
2012 Oct 30 17:21:54 user.notice awplus IMISH[20498]: show spanning-tree | include .*Role.*State.*
2012 Oct 30 17:22:02 user.notice awplus IMISH[20498]: disable
2012 Oct 30 17:22:02 user.notice awplus IMISH[20498]: show static-channel-group
```

Description

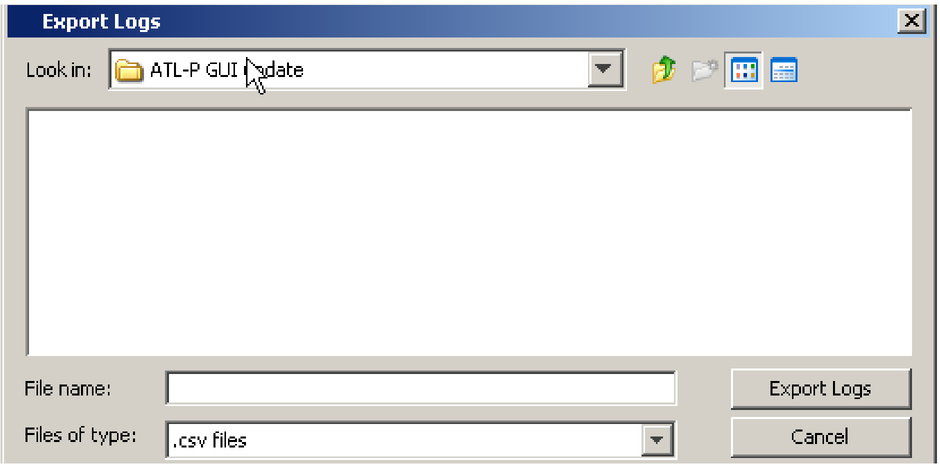
Label / Field / Button	Description
`Logs`	Display, select and export the available switch log files for troubleshooting use.
`Source`	Select the buffered log or the permanent log available on the switch to display or export to a .csv format file for use in a spreadsheet.
`Source / Refresh Log`	Select this button to display an updated buffered or permanent log.
`Source / Export Log`	Select this option to export the log to a .csv format file for use in a spreadsheet.

Management > Logs > Export Logs

The **Management** > **Logs** > **Export Logs** dialog allows you to export the switch logs as .csv format files.

Configuration Dialog

Figure C-51: Example showing the **Management > Logs > Export Logs** dialog:



Description

Label / Field / Button	Description
`File name:`	Enter the file name for the exported log file.
`Files of type:`	Select `.csv files` to export the log file as a comma separated file, so each column of the log file can be formatted in a spreadsheet.

Appendix D: Glossary

Numerics

6to4 automatic tunneling

IPv6 transition is required to migrate from IPv4 to IPv6. One method to connect to the global IPv6 network over the existing IPv4 network is called 6to4 automatic tunneling. Although this method is called '6to4 tunneling', it does not involve discrete point-to-point tunnels. The 'tunneling' in '6to4 tunneling' refers to the fact that the IPv6 packets are encapsulated in IPv4 packets to be 'tunneled' across the IPv4 domain. Hence, '6to4 tunneling' is primarily a scheme for encapsulating IPv6 packets inside IPv4 headers.

10BaseT

10 Mbps/baseband/twisted pair. The IEEE standard for twisted pair Ethernet.

802.1X

IEEE 802.1x is an IEEE Standard for port-based Network Access Control (NAC). It is part of the IEEE 802.1 group of networking protocols. It provides an authentication mechanism to devices wishing to attach to a LAN, either establishing a point-to-point connection or preventing it if authentication fails. It is used for securing wireless 802.11 access points and is based on the Extensible Authentication Protocol (EAP). Authentication is required on a per-port basis. The main components of an 802.1X implementation are:

- The authenticator - the port on this device that wishes to enforce authentication before allowing access to services that are accessible behind it.
- The supplicant - the port that wishes to access services offered by the authenticator's system. The supplicant may be a port on a PC or other device connected to this device.
- The authentication server - a device that uses the authentication credentials supplied by the supplicant, via the authenticator, to determine if the authenticator should grant access to its services.

See AAA.

For a configuration example see "Configuring 802.1X" on page 37.2. For a sample configuration script see "Sample 802.1X Authentication Configuration" on page 41.7.

A

AAA

AAA is the collective title for the three related functions of Authentication, Authorization and Accounting. These function can be applied in a variety of methods with a variety of servers.

Authentication is performed in the following contexts:

- Login authentication of user shell sessions on the console port, and via telnet/SSH.
- 802.1X authentication of devices connecting to switch ports.
- MAC authentication of devices connecting to switch ports.
- Web-authentication of devices connecting to switch ports.

Accounting is performed in the following contexts:

- Accounting of console login sessions.
- Accounting of 802.1x authenticated connections.
- Accounting of MAC authenticated connections.
- Accounting of Web authenticated connections.

There are two types of servers that can be used:

- Local user database.
- RADIUS servers.

When 802.1X authentication, MAC authentication and Web-authentication are configured to run simultaneously on a switch port this is called tri-authentication.

For more information see Chapter 41, AAA Introduction and Configuration. For a configuration example see "Configuring AAA Login Authentication" on page 41.5. For sample 802.1x, MAC authentication and Web-authentication configuration scripts see "Sample Authentication Configurations" on page 41.7.

Access-list

See ACL.

ACL

Access Control List. An ACL is one filter, or a sequence of filters, that are applied to an interface to either block, pass, or when using QoS, apply priority to, packets that match the filter definitions. ACLs are used to restrict network access by hosts and devices and to limit network traffic. See ACL sequence numbers and ACL types.

For more information see Chapter 31, Access Control Lists Introduction.

ACL sequence numbers

To help manage ACLs you can apply sequence numbers to filters. This allows you to remove filters from named and numbered ACLs without having to reconfigure an ACL. The ability to add sequence numbers to filters simplifies updates through the ability to position a filter within an ACL. When you add a new filter, you can specify a sequence number to position the filter in the ACL and you can also remove a current filter in an ACL by specifying a sequence number.

For more information see "ACL Filter Sequence Numbers" on page 31.14.

ACL types

ACLs are separated into two different types, software ACLs and hardware ACLs.

Hardware ACLs are applied directly to an interface, or are used for QoS classifications. They can be either named, or can use the following numeric ranges:

- 3000-3699 for Hardware IP ACLs
- 4000-4699 for Hardware MAC ACLs

For more information see "Defining Hardware IP ACLs" on page 31.6 and "Defining Hardware MAC ACLs" on page 31.5.

Software ACLs can be either named ACLs, using the standard or extended keyword followed by a text string, or they can use the following numeric ranges:

- 1-99
- 100-199
- 1300-1999
- 2000-2699

Software ACLs are used in features such as SNMP.

Active Master

The switch that manages the stack, or VCStack, also referred to as the Stack Master.

See Disabled Master for information about how this relates to Stack Master or Active Master.

Address resolution

The process of resolving and mapping hardware MAC addresses into their corresponding network layer IP addresses. Depending on the underlying network, address resolution may require broadcasts on a local network.

For more information see "ARP" on page D.4.

ARP

Address Resolution Protocol. ARP is used by your device to dynamically learn the Layer 2 address of devices in its networks. Most hosts also have a MAC physical address in addition to the assigned IP address. For Ethernet, this is a 6-byte, globally unique number. ARP enables your device to learn the physical address of the host that has a given IP address.

For more information see "Address Resolution Protocol (ARP)" on page 24.3.

ASCII

The *American Standard Code for Information Interchange*. A standard character-to-number encoding widely used within the computer industry.

ASIC

Application Specific Integrated Circuit. An integrated circuit (chip) manufactured to perform a specific function.

Asynchronous

Transmission in which each character is sent individually. The time intervals between transmitted characters may be of unequal length. Transmission is controlled by start and stop elements before and after each character. See "synchronous" on page D.26

Autonegotiation

Autonegotiation lets the port adjust its speed and duplex mode to accommodate the device connected to it. When the port connects to another autonegotiating device, they negotiate the highest possible speed and Duplex mode for both of them.

Autonomous system

See ASCII.

B

BIST

Built In Self Test. A mechanism that permits the device to test itself.

Blackhole route

A blackhole route is a routing table entry that does not forward packets. A blackhole route is specified as an interface with an **ip route** command. Note that a blackhole route is also called a Null route.

B-MAC

Backbone MAC address.

BPDU

Bridge Protocol Data Unit. A Spanning tree protocol initializing packet sent at configurable intervals to exchange information among bridges in the LAN.

For information on the standardized format for MSTP BPDU messages see "MSTP Bridge Protocol Data Units (BPDUs)" on page 18.17.

Bridge

A device that connects two or more networks and forwards packets between them. Bridges function at the data link layer or Layer 2 of the OSI reference model. A bridge will filter, send or flood an incoming frame, base on the MAC address of that frame.

Broadcast

One device sends out data that is intended to be received and processed by every device that it reaches.

Broadcast domain

A section of an Ethernet network comprising all the devices that will receive broadcast packets sent by any device in the domain. Separated from the rest of the network by a Layer 3 switch.

BOOTP

Bootstrap Protocol. BOOTP is a UDP-based protocol that enables a booting host to dynamically configure itself without external interventions. A BOOTP server responds to requests from BOOTP clients for configuration information, such as the IP address the client should use.

B-TAG

Backbone TAG Field.

B-VID

Backbone VLAN ID (tunnel).

B-VLAN

Backbone VLAN (tunnel).

C

CHAP

Challenge Handshake Authentication Protocol. CHAP is an authentication method used by PPP servers to validate the identity of clients. CHAP verifies the identity of the client by using a three-way handshake, and the verification is based on a shared secret by the client and the server, such as the client's password.

CIST

Common and Internal Spanning Tree. The CIST is the default spanning tree instance of MSTP, i.e. all VLANs that are not members of particular MSTIs are members of the CIST. Also, an individual MST region can be regarded as a single virtual bridge by other MST regions. The spanning tree that runs between regions is the CIST. The CIST is also the spanning tree that runs between MST regions and Single Spanning Tree (SST) entities.

For more information see "Common and Internal Spanning Tree (CIST)" on page 18.15.

Classification

In ACLs and QoS, classification is the process of filtering and marking. Filtering involves sorting your data into appropriate traffic types. Marking involves tagging the data so that downstream ports and routers can apply appropriate service policy rules. There are two reasons to classify data:

- To provide network security (security ACLs).
- To apply service quality criteria QoS.

The main application of security ACLs is to block undesired traffic. When using ACLs though QoS, the same classification and action abilities are available, but QoS has some additional fields that it can match on and also provides the ability to perform metering, marking and remarking on packets that match the filter definitions.

For more information on QoS classification see "Classifying your Data" on page 35.7.

Class maps

Class maps are among the pivotal QoS components. They provide the means that associate the classified traffic with its appropriate QoS actions. They are the linking elements for the following functions:

- Classification.
- policy mapping. See Policy maps.
- Premarking.

The relationship between a class map and a policy map can be one-to-one or many-to-one.

For more information see "Class Maps" on page 35.7.

CLI

Command Line Interface. With three distinct modes, the CLI is very secure. In User exec mode you can view settings and troubleshoot problems but you cannot make changes to the system. In Privileged exec mode you can change system settings and restart the device. You can only make configuration changes in Global configuration mode, which reduces the risk of making accidental configuration changes.

For more information see "How to Work with Command Modes" on page 1.10 and "Commands Available in each Mode" on page 1.43.

C-MAC

Customer MAC Address.

Collision domain

A physical region of a local area network (LAN) in which data collisions can occur.

Control VLAN

In EPSR, the VLAN over which all control messages are sent and received. EPSR never blocks this VLAN.

For more information see “Ring Components and Operation” on page 56.2.

CoS

Class of Service. CoS is a method for classifying traffic on a packet by packet basis using information in the type-of-service (ToS) byte to provide different service levels to different traffic. See QoS.

For more information see “CoS to egress queue premarking” on page 35.11.

Cost

An indication of the overhead required to send packets across a certain interface.

C-TAG

Customer VLAN TAG.

C-VID

Customer VLAN ID.

C-VLAN

Customer VLAN.

D

Data VLAN

In EPSR, a VLAN that needs to be protected from loops. Each EPSR domain has one or more data VLANs.

For more information see “Ring Components and Operation” on page 56.2.

Designated bridge

Each bridge or LAN in the Spanning tree, except the Root bridge, has a unique parent, known as the designated bridge. Each LAN has a single bridge, called the designated bridge, that connects it to the next LAN on the path towards the root bridge.

For an overview of spanning tree operation see “Spanning tree operation” on page 18.2.

DHCP

Dynamic Host Configuration Protocol. A method of automatically allocating IP addresses. A DHCP server holds a pool of IP addresses from which it draws individual ones as it allocates them to users when they log on.

For more information see Chapter 60, Dynamic Host Configuration Protocol (DHCP) Introduction.

DHCP leasequery

The DHCP Leasequery protocol (RFC 4388) allows a device or process, for example a DHCP relay agent, to obtain IP address information directly from the DHCP server using DHCPLEASEQUERY messages.

For more information see "Enable DHCP Leasequery" on page 60.5.

DHCP lease probing

Probing is used by the DHCP server to check whether an IP address it wants to lease to a client is already being used by another host. Probing is configured on a per-DHCP pool basis.

For more information see "DHCP Lease Probing" on page 60.7.

DHCP Option 82

Enabling the DCHP Option 82 feature on the switch allows the switch to insert extra information into the DHCP packets that it is relaying. The information is stored in a specific optional field in the DHCP packet, namely, the agent-information field, which has option ID 82.

Note that the Option 82 agent information inserted by the DHCP snooping differs from the information added by DHCP Relay. The switch cannot be configured to use both the DHCP relay agent option and DHCP snooping.

For information about the Option 82 agent information added by DHCP Relay see "DHCP Relay Agent Option 82" on page 60.9.

DHCP relay agents

DHCP relay agents pass BOOTP and DHCP messages between servers and clients. Networks where the DHCP or BOOTP server does not reside on the same IP subnet as its clients need the intermediate routers to act as relay agents.

For information on how to configure the DHCP relay agent see "DHCP Relay Agent Introduction" on page 60.8.

Disabled Master

The Disabled Master is a variant of the Stack Master or Active Master and is used with the DMM (Disabled Master Monitoring)feature. The Disabled Master has the same configuration as the Stack Master or Active Master, but has all its switchports disabled. The Disabled Master is only used is the stack separates into two stubs. By having all switchports disabled, the Disabled Master avoids potentially detrimental network connectivity problems from having two Stack Masters or Active Masters having the same configuration. The Stack Master's or Active Master's ports are unaffected by the Disabled Master's ports, so the Stack Master or Active Master continues to forward traffic normally.

For information about the Disabled Master and the Disabled Master Monitoring feature, see the Disabled Master Monitoring (DMM) section in Chapter 77, Stacking Introduction and the stack disabled-master-monitoring command in Chapter 78, Stacking Commands.

DLF

Destination Lookup Failure. DLF is the event of receiving a unicast Ethernet frame with an unknown destination address.

DMM (Disabled Master Monitoring)

The Disabled Master Monitoring (DMM) features checks the status of the Active Master via the Stack Resiliency Link. If the Active Master fails then the Disabled Master changes state to Active Master. A Disabled Master has the same configuration as the Active Master, but has all links shutdown. This change in state for the Disabled Master to become the Active Master allows traffic forwarding to continue on the VCStack.

For information about the Disabled Master and the Disabled Master Monitoring feature, see the Disabled Master Monitoring (DMM) section in Chapter 77, Stacking Introduction and the stack disabled-master-monitoring

DNS

Domain Name System. DNS allows you to access remote systems by entering human-readable device host names rather than IP addresses. DNS works by creating a mapping between a device name, such as www.alliedtelesis.com, and its IP address. These mappings are held on DNS servers. The benefits of DNS are that domain names:

- Can map to a new IP address if the host's IP address changes.
- Are easier to remember than an IP address.
- Allow organizations to use a domain name hierarchy that is independent of any IP address assignment.

For more information see "Domain Name System (DNS)" on page 24.8.

DNS Relay

DNS Relay provides the presence of a local virtual DNS server on your AlliedWare Plus^{TM} device which can service DNS lookup requests sent to it from local hosts. The DNS Relay will usually relay the requests to an external, or upstream, DNS server.

For more information see "DNS Relay" on page 24.10.

DoS

Denial of Service. A generic term for attacks that reduce or stop the operation of a network.

DSCP value

The Differentiated Services Code Point within the TOS field of an IP packet header. This is a 6-bit number in the range 0-63.

Duplex mode

See Full duplex and Half duplex.

Dynamic channel group

A dynamic channel group also known as a LACP channel group, an etherchannel, or a LACP aggregator, enables a number of ports to be dynamically combined to form a single higher bandwidth logical connection. See LACP.

For an more information see "Link Aggregation Control Protocol (LACP)" on page 20.2. For a configuration example see "Configuring an LACP Channel Group" on page 20.5.

Dynamic Link Failover

Dynamic Link Failover (Host Attach) is a versatile feature that enables devices that do not support link aggregation to form multiple active links by using Triggers and Scripts. You can customize Dynamic Link Failover to suit almost any situation, from a simple redundant backup link to multiple active links capable of basic load-sharing.

E

EAP

Extensible Authentication Protocol. EAP carries out the authentication exchange between the supplicant and the authentication server.

Etherchannel

See Dynamic channel group.

Ethernet Protection Switching Ring

See EPSR.

EPSR

EPSR (Ethernet Protection Switching Ring) operates on physical rings of switches (note, not on meshed networks). When all nodes and links in the ring are up, EPSR prevents a loop by blocking data transmission across one port. When a node or link fails, EPSR detects the failure rapidly and responds by unblocking the blocked port so that data can flow around the ring. The EPSR components are:

- EPSR domain
- Master node
- Transit node
- Ring port
- Primary port
- Secondary port
- Control VLAN
- Data VLAN

For more information and example configurations see Chapter 56, EPSR Introduction and Configuration.

EPSR domain

A protection scheme for an Ethernet ring that consists of one or more data VLANs and a control VLAN.

For more information see “Ring Components and Operation” on page 56.2.

EGP

Exterior Gateway Protocol. EGP is an obsolete protocol that has been replaced by BIST. Not to be confused with the general term exterior gateway protocol.

Egress

Outgoing packet process.

Exterior gateway protocol

A protocol that distributes routing information to devices that connect separate autonomous systems (ASCIIs).

F

FDB

Forwarding Database.

FIB

Forwarding Information Base. The RIB (Routing Information Base) populates the FIB with the best route to each destination. When your device receives an IP packet, and no filters are active that would exclude the packet, it uses the FIB to find the most specific route to the destination. If your device does not find a direct route to the destination, and no default route exists, it discards the packet and sends an ICMP message to that effect back to the source.

For more information see “RIB and FIB Routing Tables” on page 33.4.

Full duplex

When a port is in full duplex mode, the port transmits and receives data simultaneously. See Half duplex.

G

Guest VLAN

If 802.1X authentication has been configured on access ports in the network, you might still want to provide limited network access to those users whose devices do not have 802.1x supplicant enabled, or who have unrecognized authentication credentials. The mechanism to achieve this is known as a Guest VLAN. The idea is that if the users device fails 802.1X authentication, or is not even performing any 802.1X authentication, then its connection port can be put into the guest VLAN.

For more information see “Guest VLAN Enhancements” on page 39.11 and the auth guest-vlan command on page 40.8. For a configuration example see “Configuring a Guest VLAN” on page 39.3.

H

Half duplex

When a port is in half duplex mode, the port transmits or receives but not both at the same time. See Full duplex.

Hardware ACLs

See ACL types.

I

ICMP

Internet Control Message Protocol. ICMP allows networking devices to send information and control messages to other devices or hosts.

For more information see "Internet Control Message Protocol (ICMP)" on page 24.12.

ICMPv6

Internet Control Message Protocol Version 6. ICMPv6 is an implementation of ICMP for IPv6.

For more information see "The Internet Control Message Protocol (ICMPv6)" on page 26.7.

IGMP

Internet Group Management Protocol. IGMP is a communications protocol that hosts use to indicate that they are interested in receiving a particular multicast stream.

IGMP querier or router

A device in a subnetwork that is the coordinator for all multicast streams and IGMP membership information. Each subnet only has one active querier.

IGMP snooper

A device that spies on IGMP messages to create flow efficiencies by ensuring that multicast data streams are only sent to interested ports. A snooper can decide on the best path to send multicast packets at Layer 2 but does not initiate any IGMP communications.

For a configuration example see "IGMP Snooping and Querier configuration example" on page 28.5.

IGP

Interior Gateway Protocol. A routing protocol used within an autonomous system (ASCII).

Ingress

Incoming packet process.

Interior gateway protocol

See IGP.

IP directed broadcast

An IP directed broadcast is an IP packet whose destination address is a broadcast address for some IP subnet, but originates from a node that is not itself part of that destination subnet. When a directed broadcast packet reaches a switch that is directly connected to its destination subnet, the packet is flooded as a broadcast on the destination subnet. IP directed broadcast is enabled and disabled per VLAN interface. When enabled a directed broadcast packet is forwarded to an enabled VLAN interface if received on another subnet.

IP helper

The IP Helper feature allows the switch to receive UDP broadcasts on one subnet, and forward them as broadcasts or unicasts into another subnet, so a client can use an application which uses UDP broadcast (such as Net-BIOS) when the client and server are located in different subnets. The IP Helper feature forwards UDP broadcast network traffic to specific hosts on another subnet and/or to the broadcast address of another subnet. When the IP Helper feature is enabled on a VLAN interface, the UDP broadcast packets received on the interface are processed for forwarding out through another interface into another subnet.

IRDP

ICMP Router Discovery Protocol. If this feature in configured, the device sends router advertisements periodically and in response to router solicitations. ICMP Router Discovery messages let routers automatically advertise themselves to hosts.

For more information see "ICMP Router Discovery Protocol (IRDP)" on page 24.13.

I-SID

Extended Service ID.

ISP

Internet Service Provider. An organization that offers its customers access to the Internet. The ISP connects its customers using a data transmission technology, such as dial-up or DSL etc.

I-TAG

Extended Service TAG.

L

LACP

Link Aggregation Control Protocol. LACP allows bundling of several physical ports to form a single logical channel providing enhanced performance and redundancy. The aggregated channel is viewed as a single link to each switch. The spanning tree views the channel as one interface and not as multiple interfaces. When there is a failure in one physical port, the other ports stay up and there is no disruption. LACP does not interoperate with devices that use Port Aggregation Protocol (PAgP).

For an more information see "Link Aggregation Control Protocol (LACP)" on page 20.2.

LACP aggregator

See Dynamic channel group.

LACP channel group

See Dynamic channel group.

LAG

See Link Aggregation Group.

Layer 3 switch

A Layer 3 switch is an optimized combination of routing software and specialized hardware. The software uses traditional methods (static routing commands, and routing protocols) to build up a table of the best routes to network destinations, and then writes them into a set of registers in the specialized forwarding hardware. The hardware then forwards packets, based on their Layer 3 address content, at very high data rates, using the values that are written into the registers.

Link Aggregation Group

A Link Aggregation Group is a collection of bundled switch ports for an aggregated link. Link aggregation is the bonding together of two or more data channels into a single channel that appears as single logical link of higher bandwidth increasing link performance and reliability.

For an more information see "Link Aggregation Control Protocol (LACP)" on page 20.2.
For a configuration example see "Configuring an LACP Channel Group" on page 20.5

Link Local Addresses

A Link Local Address is an IP (Internet Protocol) address that is only used for communications in the local network, or for a point-to-point connection. Routing does not forward packets with link-local addresses. IPv6 requires a link-local address is assigned to each interface, which has the IPv6 protocol enabled, and when addresses are assigned to interfaces for routing IPv6 packets.

Note that link-local addresses are retained in the system until they are negated by using the no variant of the command that established them. See the ipv6 enable command for more information.

Also note that the link-local address is retained in the system if the global address is removed using another command, which was not used to establish the link-local address. For example, if a link local address is established with the ipv6 enable command then it will not be removed using a no ipv6 address command.

LLDP

Link Layer Discovery Protocol. LLDP is a Layer 2 protocol that enables Ethernet network devices, such as switches and routers, to transmit and/or receive device-related information to or from directly connected devices on the network, and to store such information learned about other devices. LLDP is a link level ("one hop") protocol; LLDP information can only be sent to and received from devices that are directly connected to each other, or connected via a hub or repeater. Advertised information is not forwarded on to other devices on the network.

For more information see Chapter 65, LLDP Introduction and Configuration.
For configuration examples see "Configuring LLDP" on page 65.11.

LLDPDU

LLDP Data Unit. See LLDP advertisements.

LLDP advertisements

LLDP transmits advertisements as packets called LLDP Data Units (LLDPDUs). An LLDPDU consists of a set of Type-Length-Value elements (TLV), each of which contains a particular type of information about the device or port transmitting it.

LLDP-MED

Link Layer Discovery Protocol Media Endpoint Discovery. LLDP-MED is an enhancement to IEEE's 802.1AB LLDP, adding media and IP telephony-specific messages that can be exchanged between the network and endpoint devices.

For more information see "LLDP-MED" on page 65.3, "LLDP-MED: Location Identification TLV" on page 65.7 and "LLDP-MED Operation" on page 65.9. For the procedure to configure LLDP-MED see "Configure LLDP-MED" on page 65.14.

Local RADIUS Server

Local RADIUS Server provides a user authentication service feature.

For more information and configuration examples see Chapter 47, Local RADIUS Server Introduction and Configuration.

M

MAC address learning

A key optimization in Ethernet switching is that the flooding of unicast traffic is minimized. This is based on switches knowing which port to forward traffic to for given destination MAC addresses. Switches achieve this by the simple process of noting on which ports packets arrive from given MAC addresses, as those will be the ports to which return packets to those MAC addresses will need to be forwarded. This process is referred to as MAC address learning.

MAC authentication

The way that MAC-based authentication works is that when the supplicant device starts sending packets, the authenticating switch will extract the source MAC address from the packets, and send a RADIUS request that uses this MAC address as the username and password in the request. See AAA and Tri-authentication.

For a sample configuration script see "Sample MAC Authentication Configuration" on page 41.8.

Master node

In EPSR, the controlling node for a domain, responsible for polling the ring state, collecting error messages, and controlling the flow of traffic in the domain.

Master node states are:

- Complete - the state when there are no link or node failures on the ring.
- Failed - the state when there is a link or node failure on the ring. This state indicates that the master node received a Link-Down message or that the failover timer expired before the master node's secondary port received a Health message.

For more information see "Ring Components and Operation" on page 56.2.

MD5

Message Digest 5 authentication algorithm.

Metering

See Policing.

Metric

The sum of all the costs along the path to a given destination. See Cost.

MLD

Multicast Listener Discovery. MLD is used to exchange membership status information between IPv6 routers that support multicasting and members of multicast groups on a network segment. Host membership in a multicast group is reported by individual member hosts, and membership status is periodically polled by multicast routers.

MLD snooping

MLD snooping is a feature whereby a Layer 2 switch listens to or "snoops" the MLD messages passing through the switch or from member hosts and multicast routers. The purpose of MLD snooping is to provide efficient Layer 2 multicast forwarding, by sending only to hosts that have expressed an interest in receiving the multicast data.

For more information see Chapter 30, MLD Snooping Introduction and Commands.

MSTI

Multiple Spanning Tree Instance. MSTP enables the grouping and mapping of VLANs to different spanning tree instances. An MST Instance (MSTI) is a particular set of VLANs that are all using the same spanning tree.

For more information see "Multiple Spanning Tree Instances (MSTI)" on page 18.12.

MSTP

Multiple Spanning Tree Protocol. MSTP is similar to Rapid Spanning Tree Protocol (RSTP) - it provides loop resolution and rapid convergence. However it also has the extra advantage of making it possible to have different forwarding paths for different multiple spanning tree instances. This enables load balancing of network traffic across redundant links. A device running MSTP is compatible with other devices running RSTP or STP.

For more information see "Multiple Spanning Tree Protocol (MSTP)" on page 18.11. For a configuration example see "Configuring MSTP" on page 18.19.

MSTP Regions

An MST region is a set of interconnected switches that all have the same values for the following MST configuration identification elements:

- MST configuration name - the name of the MST region.
- Revision level - the revision number of configuration.
- Configuration Digest - the mapping of which VLANs are mapped to which MST instances.

Each of the MST instances created are identified by an MSTI number. This number is locally significant within the MST region. Therefore, an MSTI will not span across MST regions.

For more information see "MSTP Regions" on page 18.13.

Multicast

One device sends out data that is intended to be received and processed by a selected group of the devices it reaches.

N

NAC

Network Access Control. NAC provides unprecedented control over user access to the network in order to mitigate threats to network infrastructure. NAC uses 802.1X port-based authentication with standards-compliant dynamic VLAN assignment, to assess a user's adherence to the network's security policies, and either grant authentication or offer remediation. NAC also supports alternatives to 802.1x port-based authentication, such as Web-authentication to enable guest access, and MAC authentication for end points that do not have an 802.1x supplicant. Furthermore, if multiple users share a port then multi-authentication can be used and a Guest VLAN can be configured to provide a catch-all for users without an 802.1x supplicant.

For more information see Chapter 37, 802.1X Introduction and Configuration and Chapter 39, Authentication Introduction and Configuration.

NAS

Network Access Server. A NAS is a single point of access to a remote resource. The client connects to the NAS. The NAS then connects to another resource asking whether the client's supplied credentials are valid. Based on that answer the NAS then allows or disallows access to the resource. The NAS contains no information about what resources clients can connect to or what client credentials are valid. The NAS sends the credentials the client supplied to a resource which then validates the client.

Next hop

IP routing involves forwarding packets from one router to the next, until they reach their destination. Routers do not need to know the full path to a packet's destination, they just need to know the next router to forward the packet on to. This 'next router' is referred to as the next hop of an IP route.

Nested VLAN

See VLAN ID.

NTP

Network Time Protocol. NTP is a protocol for synchronizing the time clocks on a collection of network devices using a distributed client/server mechanism.

For more information see Chapter 58, NTP Introduction and Configuration.

Null route

A null route is a routing table entry that does not forward packets. A null route is specified as an interface with an ip route command. Note that a null route is also called a Blackhole route.

O

P

PAP

Password Authentication Protocol. PAP is an authentication protocol that uses a password and is used by PPP to validate users before allowing them to access server resources. PAP transmits plain text ASCII passwords over the network so it is not secure.

Ping

Ping tests the connectivity between two network devices to determine whether each network device can "see" the other device.

Ping-of-death attack

A type of attack on a computer that involves sending a malformed or otherwise malicious ping to a network device.

Ping polling

Ping polling is used to ensure that a device is still present, live, and contactable in the network by periodically sending a packet to an IP address and waiting for a response. Configurable actions can be performed if responses are no longer arriving.

For more information see Chapter 73, Ping Polling Introduction and Configuration. For how to configure ping polling see "Configuring Ping Polling" on page 73.4.

Policing

In QoS, once you have set-up your Classification and created your Class maps, you can start conditioning your traffic flows. One tool used for traffic conditioning is the policer (or meter). The principle of policing is to measure the data flow that matches the definitions for a particular class-map; then, by selecting appropriate data rates, allocate the flows into one of three categories, Red Yellow or Green. You then decide what action to apply to the Red, Yellow and Green data. See Premarking and remarking.

For more information see "Policing (Metering) Your Data" on page 35.15.

Policy maps

Policy maps are the means by which you apply your Class maps to physical switch ports. A policy map can be assigned to several ports, but a port cannot have more than one policy map assigned to it. See QoS.

For more information see "Policy Maps" on page 35.10.

Port bit map

An efficient method for the storage of a list of ports. Each port is represented by a single bit in a 32-bit or 64-bit value.

Port mirroring

Port mirroring enables traffic being received and transmitted on a switch port to be sent to another switch port, the mirror port, usually for the purposes of capturing the data with a protocol analyzer. The mirror port is the only switch port that does not belong to a VLAN, and therefore does not participate in any other switching. Before the mirror port can be set, it must be removed from all trunk groups and all VLANs except the default VLAN.

PPP

Point-to-Point Protocol. A data link protocol used to establish a direct connection between two networking nodes. PPP can provide connection authentication and transmission encryption. PPPoE (Point-to-Point Protocol over Ethernet) is used over broadband connections as is PPPoA (Point-to-Point Protocol over ATM) with DSL.

Premarking

In QoS, premarking relates to adding QoS markers to your incoming data traffic before it is metered. QoS markers can be applied at both the link layer (within the CoS field), and at the network layer (within the DSCP field). See Policing.

For more information see "Premarking and Remarking Your Traffic" on page 35.11.

Primary port

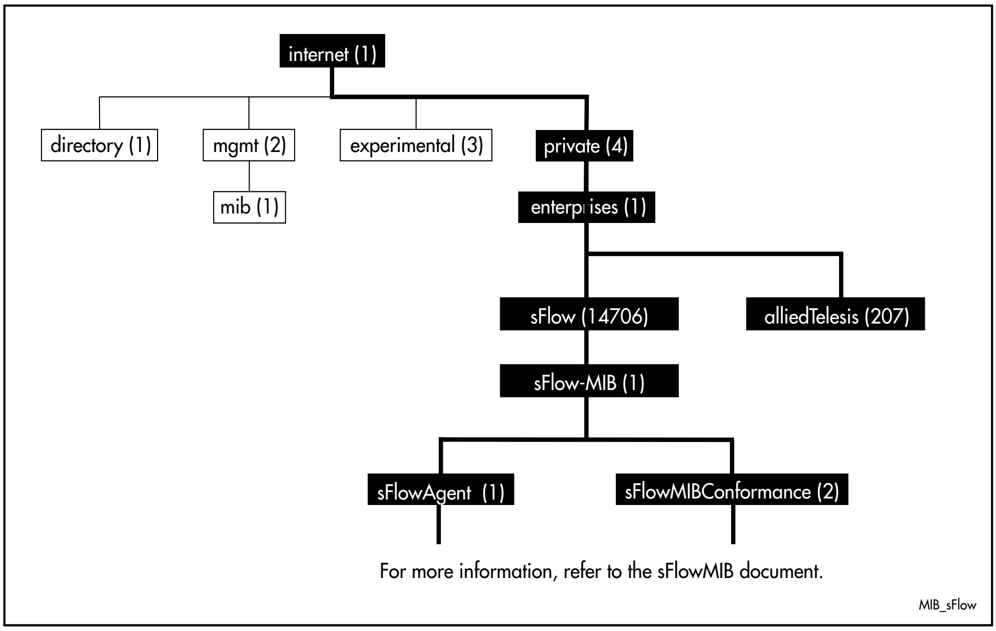
In EPSR, a ring port on the master node. This port determines the direction of the traffic flow, and is always operational.

For more information see "Ring Components and Operation" on page 56.2.

Private MIBs

In general, all objects are supported except where the relevant protocol or feature is either not supported or not applicable to the device. The following table lists the private MIBs supported by the AlliedWare Plus^{TM} Operating System. Any variations from the standard are listed.

MIB Name	Reference / Implementation
sFlow-MIB	All MIB objects are fully supported For more information, see www.sflow.org/SFLOW-MIB5.txt



Proxy ARP

Proxy ARP allows hosts that do not support routing (i.e. they have no knowledge of the network structure) to determine the physical addresses of hosts on other networks.

For more information see “Proxy ARP” on page 24.4.

PSU

Power Supply Unit.

Q

Query Solicitation

Query Solicitation minimizes the loss of multicast data after a topology change on networks that use EPSR or spanning tree (STP, RSTP, or MSTP) for loop protection. Without Query Solicitation, when the underlying link layer topology changes, multicast data flow can stop for up to several minutes, depending on which port goes down and how much of the IGMP query interval remained at the time of the topology change. Query Solicitation greatly reduces this disruption.

For more information see "Query Solicitation" on page 28.7.

QoS

Quality of Service. QoS enables you to both prioritize traffic and limit its available bandwidth. The concept of QoS is a departure from the original networking protocols, in which all traffic on the Internet or within a LAN had the same available bandwidth. Without QoS, all traffic types are equally likely to be dropped if a link becomes oversubscribed. This approach is now inadequate in many networks, because traffic levels have increased and networks often carry time-critical applications such as streams of real-time video data. QoS also enables service providers to easily supply different customers with different amounts of bandwidth. Configuring Quality of Service involves two separate stages:

- Classifying traffic into flows, according to a wide range of criteria. Classification is performed by the switch's Class maps.
- Acting on these traffic flows.

For more information see Chapter 35, Quality of Service (QoS) Introduction.

Quality of Service

See QoS.

R

RADIUS

Remote Authentication Dial-In User Service. RADIUS is a networking protocol that provides centralized AAA (Authentication Authorization and Accounting) management for clients to a network. RADIUS is a client/server protocol that runs in the application layer, using UDP (User Datagram Protocol) for data transport. RADIUS authenticates users before granting them access to network resources and can account for the usage of network resources.

For more information see Chapter 43, RADIUS Introduction and Configuration. For configuration examples see "RADIUS Configuration Examples" on page 43.14.

Redistribute

Advertise routes learnt from one routing protocol into another routing protocol.

remarking

In QoS, remarking relates to adding QoS markers to your incoming data traffic after it is metered. QoS markers can be applied at both the link layer (within the CoS field), and at the network layer (within the DSCP field). See Policing.

Remote Network MONitoring

See RMON.

Resiliency link

In VCStack, an extra, out-of-band, data link between stack members. In the event of loss of communication across the stacking connection, the stack members can determine the status of other members via communication on the resiliency link. This assists the stack members in deciding the correct course of action when communication on the stack is lost.

For more information see "Stack Resiliency Link" on page 77.13.

RIB

Routing Information Base. The RIB records all the routes that your device has learnt. Your device uses the RIB to advertise routes to its neighbor devices and to populate the FIB (Forwarding Information Base).

For more information see "RIB and FIB Routing Tables" on page 33.4.

Ring port

In EPSR, a port that connects the node to the ring. On the master node, each ring port is either the primary port or the secondary port. On transit nodes, ring ports do not have roles.

For more information see "Ring Components and Operation" on page 56.2.

For more information see "RIP" on page 32.2. For configuration examples see "Enabling RIP" on page 35.2, "Specifying the RIP Version" on page 35.4, "RIPv2 Authentication (Single Key)" on page 35.6, "RIPv2 Text Authentication (Multiple Keys)" on page 35.8 and "RIPv2 md5 authentication (Multiple Keys)" on page 35.12.

RMON

Remote Network MONitoring. RMON was developed by the IETF to support monitoring and protocol analysis of LANs with a focus on Layer 1 and 2 information in networks. RMON is an industry standard that provides the functionality in network analyzers. An RMON implementation operates in a client/server model. Monitoring devices (or 'probes') contain RMON agents that collect information and analyze packets. The probes are servers and the Network Management applications that communicate with them are clients.

For more information see Chapter 68, RMON Introduction and Configuration. For a configuration example see "RMON Configuration Example" on page 68.3.

Roaming Authentication

Roaming Authentication improves the usability of network security by enabling users to move within the network without requiring them to re-authenticate each time they move. If a supplicant (client device) moves from one wireless access point to another wireless access point, and the wireless access points are connected to different ports, then the switch (authenticator) recognizes that the supplicant has been authenticated and accepts the supplicant without requiring re-authentication.

For more information see "Roaming Authentication" on page 39.4.

Root bridge

A single Bridge is selected to become the Spanning tree's unique root bridge. This is the device that advertises the lowest Bridge ID. Each bridge is uniquely identified by its Bridge ID, which comprises the bridge's root priority (a spanning tree parameter) followed by its MAC address.

For an overview of spanning tree operation see "Spanning tree operation" on page 18.2.

Root path cost

A Spanning tree property. Each port connecting a Bridge to a LAN has an associated cost, called the root path cost. This is the sum of the costs for each path between the particular bridge port and the Root bridge. The Designated bridge for a LAN is the one that advertises the lowest root path cost. If two bridges on the same LAN have the same lowest root path cost, then the switch with the lowest bridge ID becomes the designated bridge.

For an overview of spanning tree operation see "Spanning tree operation" on page 18.2.

Route-map

A mechanism for filtering IP routes and changing their attributes.

RSTP

Rapid Spanning Tree Protocol. RSTP is an evolution of the Spanning Tree Protocol (STP) which provides for faster spanning tree convergence after a topology change. A device running RSTP is compatible with other devices running STP.

For more information see "Rapid Spanning Tree Protocol (RSTP)" on page 18.8. For a configuration example see "Configuring RSTP" on page 18.9.

S

SCP

Secure Copy Protocol. SCP allows for secure file transfer to and from the switch, protecting your network from unwanted downloads and unauthorized file copying.

For more information see "Copying with Secure Copy (SCP)" on page 6.16.

Script

A script is a sequence of commands stored as a plaintext file on a file subsystem accessible to the device, such as Flash memory. Each Trigger may reference multiple scripts and any script may be used by any trigger. When an event activates a trigger, the trigger executes the scripts associated with it in sequence. One script is executed completely before the next script begins.

See Dynamic Link Failover.

SD card

Secure Digital card.

SDHC card

Secure Digital High Capacity card.

Secondary port

In EPSR, a second ring port on the master node. This port remains active, but blocks all protected VLANs from operating unless the ring fails. Similar to the blocking port in an STP/ RSTP instance.

For more information see “Ring Components and Operation” on page 56.2.

sFlow

sFlow®[1] is an industry standard technology for monitoring high speed switched networks. It provides the ability to monitor traffic in data networks containing switches and routers.

[1] sFlow® is a registered trademark belonging to InMon Corp, San Francisco, CA.

For more information see Chapter 75, sFlow Introduction and Configuration. For how to configure sFlow see “Configuring sFlow on your Switch” on page 75.6.

sFlow agent

A network employing sFlow typically comprises a number of network (sFlow) agents that accumulate sampled data and traffic counter information. The agents then forward this data to a collector. The collector then analyses the information supplied by its agents in order to compile and display statistical profiles of the network and its traffic. The sFlow feature on your switch provides the sFlow agent capability.

For more information see “The sFlow Agent” on page 75.3.

sFlow collector

The sFlow collector receives traffic samples and counter information from a number of sFlow agents. These samples are received as a series of UDP datagrams. From the data contained within these datagrams, the collector is able to provide statistical and or graphical information of network traffic.

For more information see “The sFlow Collector” on page 75.5.

SFTP

SSH File Transfer Protocol. SFTP provides a secure way to copy files onto your device from a remote device.

For more information see “Copying with SSH File Transfer Protocol (SFTP)” on page 6.16.

Software ACLs

See ACL types.

Spanning tree

A loop free portion of a network topology. The network topology is dynamically pruned to provide only one path for any packet. See STP, RSTP and MSTP.

Spanning Tree Protocol Root Guard

See STP Root Guard.

SSH

Secure Shell. SSH is a network protocol that uses strong authentication and encryption for remote access across a nonsecure network. SSH provides sessions between a host running a SSH server and a machine with a SSH client.

For more information see Chapter 49, Secure Shell (SSH) Introduction. For how to configure a SSH server see "Configuring the SSH Server" on page 49.4. For how to configure a SSH client see "Configuring the SSH Client" on page 49.9.

stack

See VCStack.

Stack Master

The switch that manages the stack, or VCStack, also referred to as the Active Master.

See Disabled Master for information about how this relates to Stack Master or Active Master.

stack member

An individual switch that is part of a VCStack.

S-TAG

Service VLAN TAG.

Static aggregator

See Static channel group.

Static channel group

A static channel group, also known as a static aggregator, enables a number of ports to be manually configured to form a single logical connection of higher bandwidth. By using static channel groups you increase channel reliability by distributing the data path over more than one physical link.

Storm-control

Storm-control enables you to specify the threshold level for broadcasting, multicast, or destination lookup failure (DLF) traffic for a port. Storm-control limits the specified traffic type to the specified threshold.

For more information see "Storm-control" on page 14.11.

Storm protection

Storm protection uses QoS mechanisms to classify on traffic likely to cause a packet storm (broadcast and multicast). With QoS storm protection, several actions are possible when a storm is detected:

- You can disable the port physically.
- You can disable the port logically.
- You can disable the port for a particular VLAN.

For more information see "Storm Protection" on page 35.23.

STP

Spanning Tree Protocol. STP is the original bridge protocol defined by IEEE standard 802.1D-1988. It creates a single spanning tree over a network.

For more information see "Spanning Tree Protocol (STP)" on page 18.5. For a configuration example see "Configuring STP" on page 18.6.

STP Root Guard

Spanning Tree Protocol Root Guard. STP Root Guard designates which devices can assume the role of Root bridge in an STP network. This stops an undesirable device from taking over this role, where it could either compromise network performance or cause a security weakness.

See the spanning-tree guard root command on page 19.44.

Subnet address

A subnet portion of an IP address. In a subnetted network, the host portion of an IP address is split into a subnet portion and a host portion using an address or subnet mask.

Subnet mask

A bit mask used to select bits from an Internet address for subnet addressing. The mask is 32 bits long and selects the network portion of the Internet address and one or more bits of the local portion. Sometimes called address mask.

Switch instance

A single switch chip with its associated ports, internal data interfaces, hardware tables, and packet buffer memory.

S-VID

Service VLAN ID.

S-VLAN

Service VLAN.

synchronous

Transmission in which the data characters and bits are transmitted at a fixed rate with the transmitter and receiver synchronized. This eliminates the need for start-stop elements, as in asynchronous transmission, but requires a flag character to be transmitted when there is no data to transmit. See Asynchronous.

T

TACACS+

TACACS+ (Terminal Access Controller Access-Control System Plus) provides a method for securely managing multiple network access points from a single management service. TACACS+ is a TCP-based access control protocol that allows a device to forward a user's username and password to an authentication server to determine whether access can be allowed. In addition to this authentication service, TACACS+ can also provide authorization and accounting services. One of the features of TACACS+ is the ability to separate authentication, authorization and accounting so that these functions can be provided independently on separate servers.

For information on the AlliedWare Plus implementation of TACACS+, see Chapter 45, TACACS+ Introduction and Configuration and Chapter 46, TACACS+ Commands.

TCN

Topology Change Notification.

Thrash limiting

MAC address thrashing occurs when MAC addresses move rapidly between one or more ports or trunks, for example, due to a network loop. Thrash limiting enables you to apply actions to a port when thrashing is detected. It is supported on all port types and also on aggregated ports.

For more information see “Thrash Limiting” on page 14.13

TLV

Type-Length-Value. A single LLDPDU contains multiple TLVs. TLVs are short information elements that communicate complex data, such as variable length strings, in a standardized format. Each TLV advertises a single type of information, such as its device ID, type, or management addresses. See LLDP advertisements.

Traceroute

Traceroute is used to discover the route that packets pass between two systems running the IP protocol. Traceroute sends an initial UDP packets with the Time To Live (TTL) field in the IP header set starting at 1. The TTL field is increased by one for every subsequent packet sent until the destination is reached. Each hop along the path between two systems responds with a TTL exceeded packet (ICMP type 11) and from this the path is determined.

Transit node

In EPSR, nodes other than the master node in the domain.

Transit node states are:

- Idle - the state when EPSR is first configured, before the master node determines that all links in the ring are up. In this state, both ports on the node are blocked for the data VLAN. From this state, the node can move to Links Up or Links Down.
- Links Up - the state when both the node's ring ports are up and forwarding. From this state, the node can move to Links Down.
- Links Down - the state when one or both of the node's ring ports are down. From this state, the node can move to Preforwarding.
- Pre-forwarding - the state when both ring ports are up, but one has only just come up and is still blocked to prevent loops. From this state, the transit node can move to Links Up if the master node blocks its secondary port, or to Links Down if another port goes down.

For more information see "Ring Components and Operation" on page 56.2.

Tri-authentication

Authentication commands enable you to specify three different types of device authentication: 802.1X authentication, MAC authentication, and Web-authentication. All three types can be configured to run simultaneously on a switch port. The simultaneous configuration and authentication of all three types on a port is called tri-authentication.

For a configuration example see "Tri-Authentication Configuration" on page 39.2.

Trigger

A trigger is an ordered sequence of scripts that is executed when a certain event occurs. Each trigger may reference multiple scripts and any Script may be used by any trigger. When an event activates a trigger, the trigger executes the scripts associated with it in sequence. One script is executed completely before the next script begins.

See Dynamic Link Failover.

Type-Length-Value

See TLV.

U

unicast

Two individual devices hold a conversation just between themselves.

V

VCStack

A group of two or more switches operating as a single switch. See Virtual Chassis Stacking.

VCStack Fast Failover

VCStack Fast Failover provides absolutely minimal network downtime in the event of a problem with the stack.

See the reboot rolling command on page 78.7.

VID

VLAN Identifier or VLAN ID. When you create a VLAN you give it a numerical VID which is included in VLAN-tagged Ethernet frames to and from this VLAN.

Virtual Chassis Stacking

Virtual Chassis Stacking (VCStack™) is the name given to two or more Allied Telesis switches that are configured to operate as a single switch. From a configuration and management point of view, it is as though the switches are just one device with a seamless transition from the ports of one stack member to the ports of the next.

For more information see Chapter 77, Stacking Introduction.

VLAN classification

A packet can be allocated VLAN membership based on its protocol, subnet, or port.

VLAN ID

See VID.

VLAN Identifier

See VID.

VLAN stacking

See VLAN ID.

VLAN tag

IEEE standard 802.1q defines an additional 4 byte tag field that can be inserted immediately following the MAC address, plus any routing fields present. This field contains a 12 bit VLAN identifier, commonly referred to as the VLAN tag. The VLAN tag is used to determine which VLAN a given frame should be forwarded to.

Other tags included in the 802.1q tag field is a Tag Protocol Identifier tag, and a Type of Service tag used to determine data priority.

Voice VLAN

Voice VLAN automatically separates voice and data traffic into two different VLANS. This automatic separation places delay-sensitive traffic into a voice-dedicated VLAN, which simplifies QoS configurations.

For more information see "Voice VLAN" on page 65.3.

VoIP

Voice over Internet Protocol. Enables the delivery of voice communications over IP networks such as the Internet or other packet-switched networks instead of over traditional telephony circuits.

VRID

Virtual Router Identifier.

VRRP

Virtual Router Redundancy Protocol. VRRP combines two or more physical switches into a logical grouping called a virtual router. The physical switches then operate together to provide a single logical gateway for hosts on the LAN. If the master fails, the other devices assume the virtual IP address.

For more information see Chapter 54, VRRP Introduction and Configuration. For configuration examples see "Configuration examples" on page 54.9.

W

Web-authentication

The switch sends a login screen to the client webbrowser which must be authenticated before access is granted to the network. See AAA and Tri-authentication.

For a sample configuration script see "Sample Web-Authentication Configuration" on page 41.9.

Wildcard mask

A subnet mask in which bits set to 0 indicate an exact match and bits set to 1 indicate 'don't care'.

X

XEM

High Speed Expansion Module.